TypeTool® 2.0

for Macintosh®

User Manual

Copyright ©1992-2003 by FontLab, Ltd. All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written consent of the publisher. Any software referred to herein is furnished under license and may only be used or copied in accordance with the terms of such license.

FontLab, *FontLab logo*, *ScanFont*, *TypeTool*, *SigMaker, AsiaFont Studio, FontAudit* and *VectorPaint* are either registered trademarks or trademarks of FontLab, Ltd. in the United States and/or other countries.

Apple, the *Apple Logo*, *Mac*, *Mac OS*, *Macintosh* and *TrueType* are trademarks of Apple Computer, Inc., registered in the United States and other countries.

Adobe, *PostScript, Type Manager* and *Illustrator* are trademarks of Adobe Systems Incorporated which may be registered in certain jurisdictions.

Windows, *Windows 95*, *Windows 98, Windows XP* and *Windows NT* are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

IBM is a registered trademark of International Business Machines Corporation.

Macromedia, Fontographer and *Freehand* are registered trademarks of Macromedia, Inc.

Other brand or product names are the trademarks or registered trademarks of their respective holders.

THIS PUBLICATION AND THE INFORMATION HEREIN IS FURNISHED AS IS, IS SUBJECT TO CHANGE WITHOUT NOTICE, AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY FONTLAB, LTD.

FONTLAB, LTD. ASSUMES NO RESPONSIBILITY OR LIABILITY FOR ANY ERRORS OR INACCURACIES, MAKES NO WARRANTY OF ANY KIND (EXPRESS, IMPLIED OR STATUTORY) WITH RESPECT TO THIS PUBLICATION, AND EXPRESSLY DISCLAIMS ANY AND ALL WARRANTIES OF MERCHANTABILITY, FITNESS FOR PARTICULAR PURPOSES AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.

User manual release 2.0 [03/2003]

Contents

CONTENTS 3

INTRODUCTION 9

About this Manual 10

System Requirements 11

TYPETOOL USER INTERFACE 13

Basic Terms 14

Getting Started 18

Customizing TypeTool's User Interface 20
- Customization of the Keyboard 21
- Links to External Programs 23

TypeTool Windows 25
- Font Window 26
- Glyph Window 28
- Metrics Window 31

Panels 34

EDITING FONTS 35

Opening Fonts 36
- Font Formats 39
- Import Options 41

Creating a New Font 45

Font Window 46

Encoding Modes 50
- Characters, Codes and Glyphs 51
- Names Mode 55
- The Unicode Standard 57
- Codepages 58

Using the Font Window 62
- Navigating 63
- Selecting 64

Context Menu 65

Moving Glyphs 67

Saving the Font 69
Autosave 70

Copying and Pasting Glyphs 71
Copying Glyphs to Another Font 72
Copying Composite Glyphs 72
Duplicating Unicode Indexes 73

Deleting Glyphs 74

Creating New Glyphs 75

Searching for Glyphs 76

Renaming Glyphs 78

The Font Map Panel 80
Managing Double-Byte Codepages 82

Applying Actions 83

Exporting Fonts 84
Font Formats 84
Exporting for PC 85
Exporting for Macintosh 86
Export Options 94
Type 1 Export Options 95
TrueType Export Options 98
Recommended Export Options 100

Printing Fonts 101

Font Window Options 102
View Options 102
Tuning Options 103

THE FONT HEADER 105

Font Info Dialog Box 106

Font Names 108
Basic Identification and Names 109
Copyright Information 111
Designer Information 112
License Information 113

Font Identification 114
Version Information 114
Basic Font Identification 115

Metrics and Dimensions 117
Font UPM Value 117
Basic Font Dimensions 118
Advanced Vertical Metrics 120

Encoding and Unicode 123
Supported Codepages 124

THE GLYPH WINDOW 127

Glyph Window Contents 128

Selecting a Glyph for Editing 130

Changing the View in the Glyph Window 131
Quick Zoom Selection 132

Bezier Drawing Tools 134

Editing Layers 136

Outline Layer 137
Contours 139
Outline Appearance 144
Moving Nodes 148
Using the Keyboard 151
Non-node editing 152
Changing Connection Type 154
Deleting Nodes 155
Deleting Lines and Curves 155
Eraser Tool 156
Inserting Nodes 157
Meter Tool 159
Using the Drawing Tool 161
Adding Points to a Contour 162
Converting Primitives 163
Breaking and Joining Outlines 164
Node Commands 165
Node Properties 168

Selections 170
Using the Magic Wand Tool 171
Moving the Selection 172
Copying the Selection 172
Transform Selection 175
Selection Commands 179
Selection Properties Panel 180
Transformation Panel 181
Building an Outline From Blocks 182
Outline Actions 186
Merging and Intersecting Contours 187

Metrics 188
Editing Metrics 189
Baseline Properties Panel 190
Metrics Properties Panel 190

Grid Layer 191

Guidelines Layer 192
Editing Guidelines 193
Guidelines Popup Menu 195
Guidelines Properties Panel 195

Hints Layer 196
Editing Hints 197
Hint Popup Menu 198
Hint Commands 198
Hint Properties Panel 198

Vertical Metrics 199

VectorPaint Tools 200
Pen (Contour) Tool 202
Brush Tool 203
VectorPaint Options 205
Line Tool 206
Ellipse and Rectangle Tools 206
Text Tool 207

Working with Composite Glyphs 208
Decomposing 209
Component Positioning 210
Component Properties 212

Importing and Exporting Glyphs 213

Printing a Glyph 215

EDITING FONT METRICS 217

What are Font Metrics? 218
Kerning 219
Metrics Files 220

Metrics Window 221
Selecting a String for Previewing or Metrics Editing 223

Editing Metrics 226
Manual Metrics Editing 227
Using the Keyboard 228
Using the Metrics Table 229
Automatic Metrics Generation 230

Editing Kerning 232
Manual Kerning Editing 232

Using the Keyboard 233
Using the Table 233
Automatic Kerning Generation 234
Resetting Kerning 236

Opening Metrics Files 237

Saving Metrics Files 239

Printing a Sample String 240

TRANSFORMATIONS 241

The Transform Dialog Box 242

Transformation Actions 244
Outline Transformation 245
Hints and Guidelines Transformation 249
Metrics Transformation 250

INDEX 251

Introduction

Thank you for purchasing TypeTool 2 – a new version of our entry-level font editor.

The key features of TypeTool 2 are:

- Outline editors with more than 10 tools and 200-level undo/redo
- Import and export of fonts in TrueType and PostScript Type 1 formats with up to 65000 characters
- Import and export of individual glyphs in EPS format
- Metric and kerning editing module with customizable autospacing and autokerning features
- Import and export of font metrics files in PFM and AFM format
- Automatic Type 1 and TrueType hinting
- Automatic transformation of glyphs
- VectorPaint tools
- Support of two encoding modes and an unlimited number of encoding tables
- Easy-to-use drag/drop-based user interface
- Popup menus and property panels everywhere
- Sample printing of fonts and individual glyphs
- Smooth outline preview

About this Manual

This manual covers the Macintosh version of TypeTool 2.0.

The following chapters describe all of TypeTool's features in full detail. They are organised to cover all the functions in their usual sequence.

TypeTool User Interface

This chapter covers the basic definitions of the TypeTool user interface and its customization and gives a short description of all the TypeTool editing windows and panels.

Editing Fonts

This chapter explains how to modify fonts, copy characters, change encoding tables, select characters for editing, and export fonts.

The Font Header

This chapter provides a detailed description of the Font Header data and the TypeTool tools intended to manage it.

The Glyph Window

TypeTool includes powerful outline-editing tools that are described in this chapter.

Editing Font Metrics

If you want to create a professional-looking font you have to edit the font's metric data. The glyphs' widths, sidebearings, and kerning can be edited in TypeTool automatically or manually. This chapter shows you how.

Transformations

From scale to autohinting – transformation filters can help you instantly expand your font collection. This chapter gives detailed descriptions of all the transformations and their usage in TypeTool.

System Requirements

The Macintosh version of TypeTool requires the following hardware and software configurations:

- a Power PC based computer capable of running Mac OS 9.1;
- one of the following versions of Mac OS:

  Mac OS 9.1 or later classic operating system (v 9.2.2 is recommended) with CarbonLib 1.4 or later or

  Mac OS X 10.1 or later (v 10.2 is recommended).

At least 14Mb of free space on the HDD and at least 64 Mb RAM. TypeTool will start on 32Mb RAM but you will need more RAM to open bigger fonts.

A Note on Intellectual Property

Digital fonts are complex computer programs created with a good deal of hard work by individuals and companies. They are valuable intellectual property and are protected by trademark, copyright, and patent laws. The details and extent of this protection vary in different countries, but the basics are as follows:

Trademark: A font name (and only the name) may be trademarked. Only the trademark owner or licensees may use the name to describe a font.

Copyright: Computer programs are copyrighted. In the U.S.A. this happens automatically as soon as the program is written. Further rights may be secured by registering the copyright.

Patent: Some fonts, if they are distinctive and unusual enough, may be granted a design patent. Only the patent holder or licensees may use this font design.

If you purchase a font and then modify it for your own use you are probably within the bounds of “fair use” and the font licensing agreement. However, if you modify a purchased font and then sell or distribute it you may be in violation of copyright, patent or licensing laws. Please read your font license agreement carefully or contact the licenser to determine your rights and obligations.

TypeTool User Interface

Before we start talking about fonts and the TypeTool font-editing features let's spend some time learning the TypeTool user interface. For the most part it is a standard Macintosh interface so if you know how to navigate in Mac OS you will feel comfortable with TypeTool. In other parts it is unique and that is where we will focus.

Some of the interface elements in TypeTool 2 are completely customizable and from this chapter you will learn how to change the TypeTool interface so it will best fit your needs.

Please note that further in the book we will refer to menu commands, toolbar buttons and keyboard shortcuts as they appear in the default TypeTool environment under Mac OS X v 10.2, prior to any modifications you may make. All screenshots for the book illustration were also made under Mac OS X v 10.2.

1

Basic Terms

We cannot go any further without defining a few terms that are critical to understanding TypeTool and fonts in general.

Character

The minimal unit of the written language – a part of the alphabet, a symbol.

Any picture that can be recognized as having the same meaning represents the same character:

All the pictures above mean the character 'A'

Please note that sometimes pictures that look the same represent different characters:

A Latin 'A'

A Cyrillic 'A'

A Greek 'Alpha'

Characters have *codes* that are used to store text data on a computer.

Glyph

The basic element of the font, literally – an image that is printed. All glyphs are unique, even if they represent the same character.

Glyphs are used to represent characters. Please note that many different glyphs may be used to represent the same character, even in the same font:

Font

An organized collection of glyphs and font header information. Usually glyphs that are united in a font have some similarities in design and other properties.

In the past, a “font” was defined as a single size of the characters of a particular typeface. Now, since fonts are scalable, the term “font” covers all possible sizes of the same typeface design.

Encoding

When text is printed an important process takes place: character to glyph mapping. The source text (in computer form) is a list of codes that represents a list of characters. A font (see above) is a collection of glyphs. So there must be some way to relate characters to glyphs so that when the computer’s operating system encounters a certain character it knows which glyph to print. This “mapping” (or “vector”) is called the encoding. Sometimes the encoding information resides within the font itself as part of the header and other times it is in a separate file.

Font Family

It is important to know the difference between a font and a font family. A font family is a set of fonts that represents some design idea. “Times” is a font family (sometimes called typeface). “Times Bold Italic” is a font.

A font family may include from one to a few dozen fonts.

Glyph name

The only identification of a glyph (other than its visual appearance) is its name. A Western glyph name consists of Latin characters, digits and punctuation. It is highly recommended you name glyphs in accordance with the following rules:

1. No spaces.
2. No digits at the beginning.
3. Only '.' And '_' punctuation marks are allowed in the name.

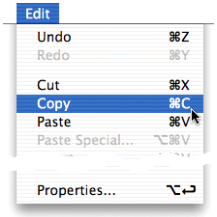
Menu

When we refer to menu items in the main TypeTool menu, we will use the following notation:

[top menu item]>[sub-item]

For example:

Edit>Copy means: click the word Edit on the menu bar and select the **Copy** command from the menu:



Directories

When we have to refer to one of the folders created during TypeTool installation, we will use the following syntax:

`TypeTool/[directory name]`

Where TypeTool is the name of the folder where you have installed TypeTool 2 (usually /Applications/TypeTool2) and [directory name] is the name of the folder, as in the example below:

`TypeTool/Data`

Mouse

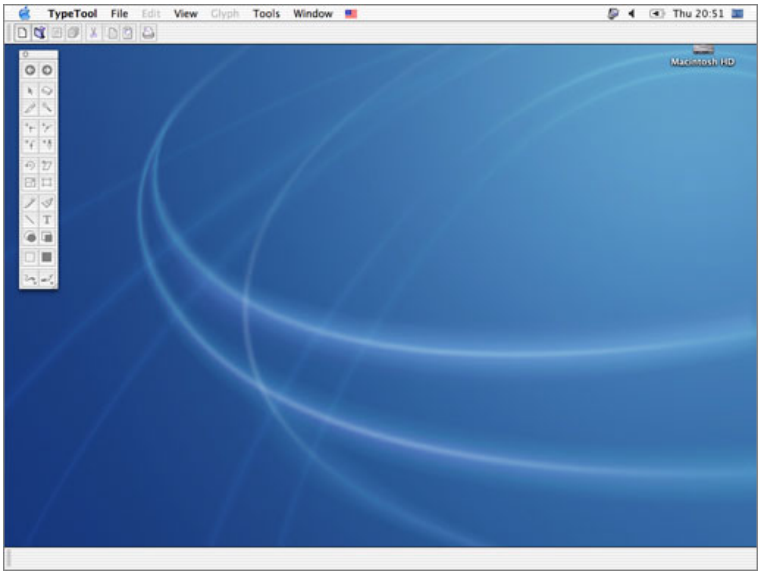
Click the mouse on some object	Position the mouse cursor on the object and click the mouse button
Ctrl-click something	Position the cursor over "something", hold down the **CTRL** key on the keyboard and click the mouse button. If you have a mouse with two buttons, you may use the right mouse button instead of pressing the **CTRL** key.
Drag some object	Position the cursor on the object, press the mouse button and move the mouse to move the object. Release the mouse button when you're done.

Context Menu

Most windows and panels in TypeTool have attached context menus. To open the context menu, **CTRL**-click (or click the right mouse button if present) an empty area in the window or panel.

Getting Started

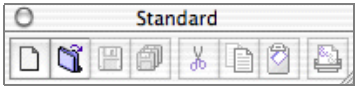
When you run TypeTool 2 for the first time (to run TypeTool double-click on its icon) you will see a welcome screen for a few seconds and then the TypeTool window:



Like almost all Macintosh programs TypeTool has a menu, a few toolbars and a status bar at the bottom.

There is nothing special about the TypeTool menu except that there is a difference between the Classic and Mac OS X environments. In Mac OS X there is an **Application** menu containing the **Quit** and **Preferences** commands. Under Mac OS 9 the **Quit** command traditionally resides in the **File** menu while the **Preferences** command resides in the **Tools** menu.

The usual location of toolbars is at the top of the screen, but if you want to put them somewhere else, just drag them there:



The status bar can be placed only at the bottom (default) or at the top of the screen. Some tool specific toolbars are floating only and cannot be docked to the sides of the screen.

You can easily choose which toolbars you want to see: use the **Toolbars** submenu in the **View** menu.

Following is a list of all common toolbars with a few comments about each:

Standard	Contains basic commands like file open and save, copy/paste and print
Tools	Probably the most important – a toolbar containing the main editing tools and frequently used operations
Show Layers	Controls the appearance of basic *Editing layer* in *Glyph window*. It is analogous to the **View>Show Layers** menu.

You may notice a few *italic* terms. We will describe them later. Specifically, *Glyph window* will be introduced in a few pages and *Editing layers* in the "Glyph Window" chapter.

OK, we are almost ready to open a sample font, but before we do let's talk about *customization* of the TypeTool user interface.

Customizing TypeTool's User Interface

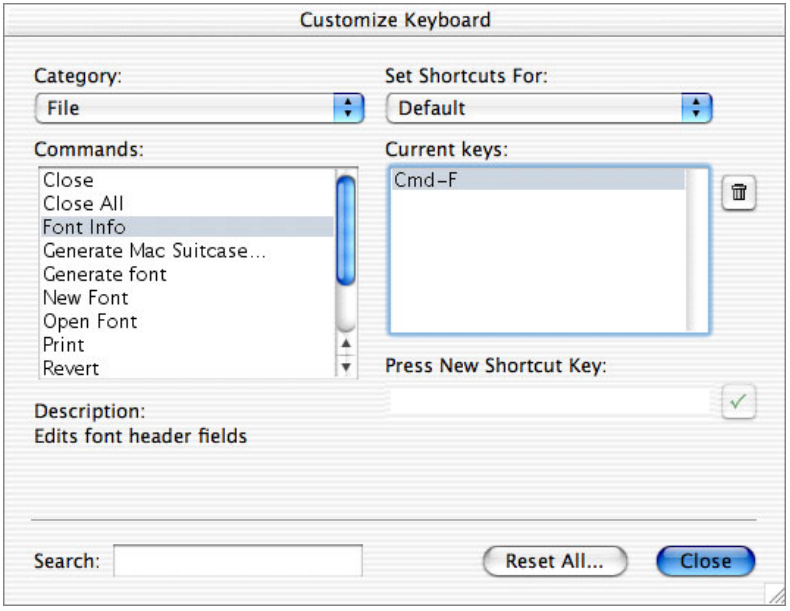
As you may infer from the title of this section some of the TypeTool user interface (which means the toolbars and keyboard shortcuts) is customizable. We believe our default interface is the easiest to use, but if for some reason you don't like it, you are free to make any changes you want. If you don't want to change anything in the TypeTool user interface, you can fast forward to the next section.

You cannot customize menus in TypeTool 2 for Macintosh yet. The customization of toolbars is also restricted to switching them on or off or arranging them on the screen.

The general idea of keyboard customization is simple: there is a long list of commands that you can use with keyboard shortcuts. Through customization you can assign any command to a combination of keys pressed on the keyboard.

Customization of the Keyboard

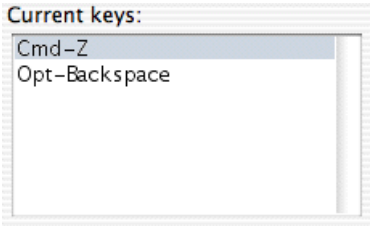
Select the **Tools>Customize** command and the **Customize Keyboard** dialog box will appear:



In the left area of the page you can select the command that you want to customize. Choose the commands category in the top popup menu and the command itself in the list below. You will see a short command description below the commands list.

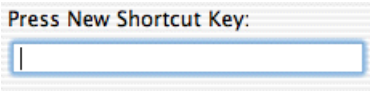
To easily find a particular command by its name, type it in the **Search** field at the bottom of the dialog box.

On the right part of the page you will see the list of keyboard shortcuts currently defined for that command:



The button at the right of the list allows removal of one of the existing shortcuts.

To define a new keyboard shortcut, select a command and position the cursor on the editing field below the **Press New Shortcut Key** label:



When the editing field has focus just press the combination of keys that you want to assign. A description of that combination will appear in the editing field and you can click the **Assign** button **to assign that combination to the currently selected command**.

Press the **Reset All** button **to reset all changes back to TypeTool defaults**.

Press the **Close** button at the bottom after you finish customizing the keyboard shortcuts to save changes and close the dialog box.

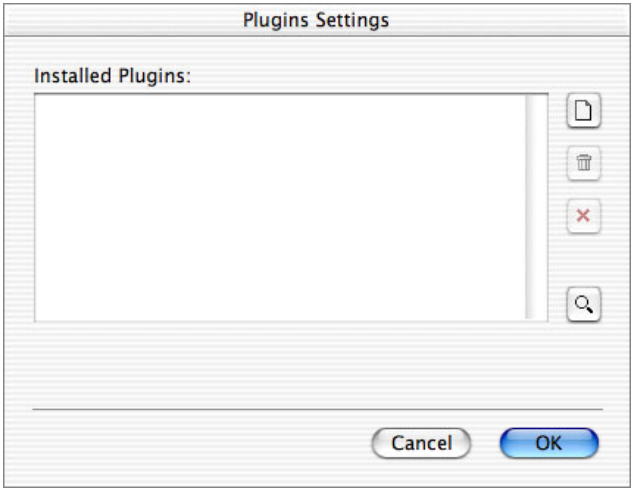
Links to External Programs

Some applications available from FontLab Ltd. have a common communication interface, which allows them to easily exchange font related data. We call programs that can accept data from other programs *FontLab Server Applications*. Similarly, programs that can send data to FontLab servers will be referred to as *FontLab Client Applications* or plugins. And programs that combine server and client capabilities are called *FontLab Client/Server Applications*.

TypeTool itself can work as server and sometimes as client as well. For example, it can accept outline data from BitFonter (our bitmap font editor) or send outlines to BitFonter for rasterization.

To make TypeTool communicate with plugins installed locally and to have instant access to them from the TypeTool menu, you have to show TypeTool where they are.

Use the **Tools>External Tools>Settings** command to find other applications that can work together with a FontLab Server. The Plugins Settings dialog box appears:



There is a list of the installed plugins in the dialog and it is empty by default. Click this button: **to add a link to the program that you know to be a plugin**. Select the plugin application in the standard Open File dialog box and press **Open**. The application name will appear in the list.

An easier way is to let TypeTool find the plugins itself. Press the button at the right of the list and TypeTool will automatically search for plugins on the available disks.

The button at the right of the list allows removal of the link to the plugin selected in the list.

Press the **Reset All** button **to clear the list of plugins**.

After you press the **OK** button the plugins added to the list will appear in the **Tools>External Tools** menu to let you quickly launch them when needed.

TypeTool Windows

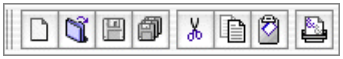
There are only three types of Windows in TypeTool:

Font window	Represents one of the opened fonts
Glyph window	Used to edit glyphs
Metrics window	Used to edit glyph metrics and kerning.

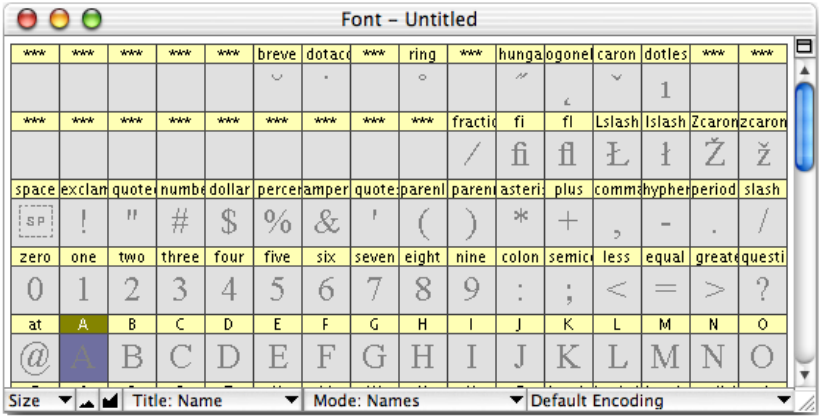
In this chapter we will provide only very basic information about the main windows. Please refer to the “Editing Fonts”, “Glyph Window” and “Editing Font Metrics” chapters to get detailed information about the windows and their features.

Font Window

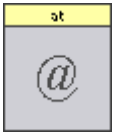
As an exercise let's create a font in order to demonstrate the TypeTool Windows. Use the **New** command in the **File** menu or click this button on the Standard toolbar:



You will see the Font window:



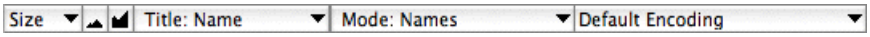
As you can see, this window has a footer with a few buttons and options and a big table of cells that represent characters and glyphs. Each cell has a caption that contains glyph identification information: name, Unicode, index or some other data:



Cells can also contain little icons that show properties of glyphs, but more about that later.

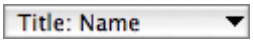
There are no glyphs in the font that we just created, but the Font window nevertheless shows some pictures in the glyph cells. These are template images that show which character should be placed in the cell. TypeTool has templates for thousands of characters, so you will usually know where to place new characters.

We'll discuss navigation in the Font window later, in the Editing Fonts chapter, so let's talk about the Font window command bar, which is located at the bottom of the window:

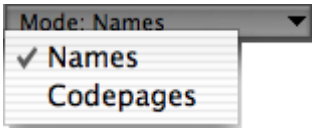


On the command bar you can easily find a popup and two zoom buttons on the left, which allowyou to change the cells size and are described later in this manual.

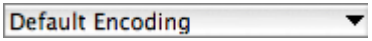
Right of the buttons there is a popup menu, which allows you to change the information that appears in the character cell's captions:



At the right of the **Title** popup menu there is a popup that allows a choice of encoding modes. Again, a detailed description of this follows. Just a few words here: any glyph in the font may be identified by a name or Unicode index. Two items in the popup menu allow you to choose one of two modes: Names or Codepages:



Next is a popup menu that allows you to change the encoding table or codepage for the current font:



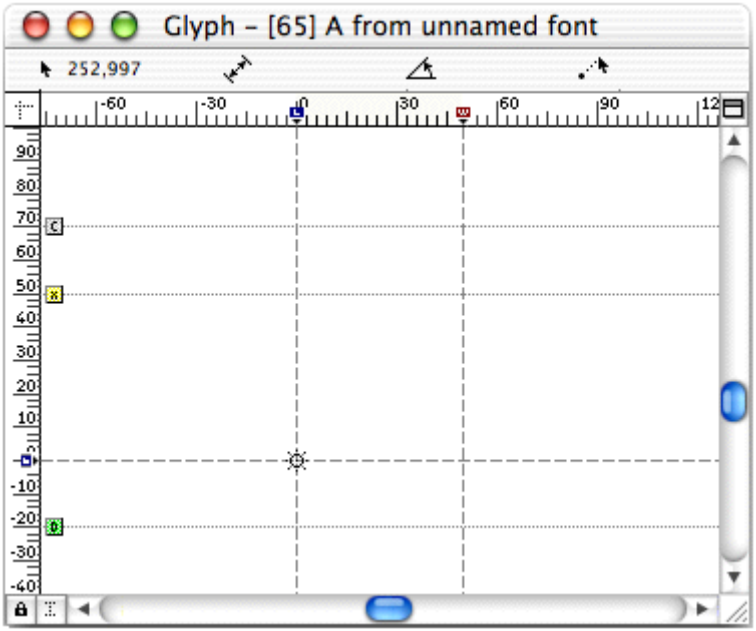
We will talk about encodings later, but you could choose a couple different ones from the popup menu and see how the Font window changes.

That's all about the Font window for now so let's open the Glyph window.

Glyph Window

To open a Glyph window for editing individual glyphs you need to create one. Remember, we started with a new font that doesn't have any glyphs. **To create a glyph**, double-click on any cell in the Font window. You will see that the gray cell (which means there is no glyph defined) is replaced by a white one, which represents a glyph that is defined, but contains no image. When you draw or paste something into it, the white cell will show a small picture of the glyph.

After the glyph cell is created we are ready to open the Glyph window. Select the glyph cell (just click on it with the mouse button) and double-click it to open the Glyph window. It will immediately appear on screen:



Instead of double-clicking, you can also use several other methods **to open the Glyph window**:

1. **Ctrl**-click the glyph cell and select the **Edit in new window** command in the context menu.
2. Select the glyph and choose **New Edit window** in the **Window** menu.
3. And finally, select the glyph cell and just press the **Enter** key on the keyboard.

If you have more than one glyph in your font (which is normal when you open an existing font) and have a glyph window already open when you double-click another glyph in the Font window (or use any other method of opening a glyph window except the **New Edit window** command or a button on the toolbar), a new glyph will appear in the original glyph window. If you need to open many glyph windows simultaneously just hold down the **Cmd** key when you double-click the new glyph cell or otherwise open a new Glyph window.

You may have as many open glyph windows as you want, just close those you don't need so as not have all your workspace covered with glyph windows. You may use the **Cmd-W** shortcut to close windows. If the current window is the Font window closing it will close the font.

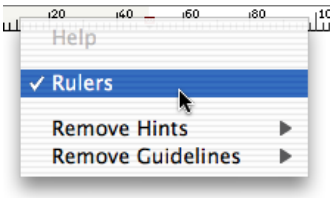
Glyph Window Contents

All windows in TypeTool have a similar layout: control panel on the top and main area covering most of the window. The glyph window is no exception. Click on the small button on the top of the vertical scroll bar: and the Glyph window control area will appear. The control area contains zoom selection tools, a popup menu, and a few toolbar buttons:

They are used to select the zoom mode of the Glyph window and to choose the **Zoom in** and **Zoom out** commands and **Undo** or **Redo** commands.

The main area of the window has scroll bars to change the view of the glyph, and vertical and horizontal ruler bars.

You can switch the ruler bars on and off with the **Rulers** option in the **View** menu. A quicker way is to **Ctrl**-click the ruler and choose the option in the context menu:



At the bottom-left corner of the Glyph window you will find two buttons, Lock and Meter:

The Lock button controls quick access to the font glyph – when it is in the "unlocked" state you can use the keyboard to directly access the glyphs. I.e. when you press a key the corresponding glyph will automatically open in the glyph window.

The Meter button controls the appearance of the Meter bar, which sits above the glyph window ruler and shows the current coordinates and other parameters of the cursor:

We will return to a more detailed description of the Glyph window properties in the "Glyph Window" chapter.

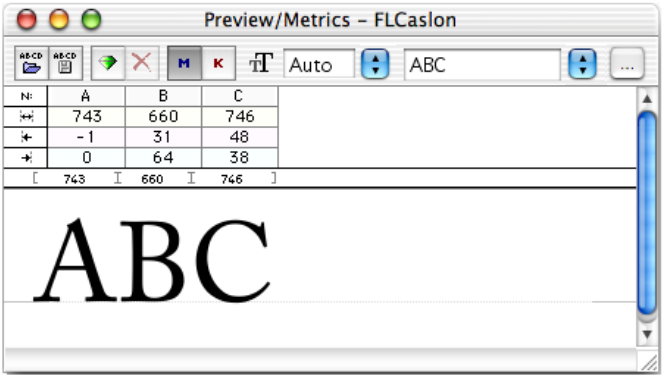
Finally, let's quickly preview the last window in TypeTool: the Metrics window.

Metrics Window

The Metrics window is used to adjust glyph metrics – glyph sidebearings and kerning. Open any existing font from the disk, e.g. FREESERF.VFB from the TypeTool/Samples folder.

To open the metrics window select some glyphs in the Font Window and click on the **New Metrics window** command in the **Window** menu (**CMD-OPTION-SHIFT-M** shortcut).

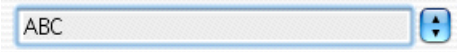
You will see a new window:



Glyphs that are currently selected in the Font window or the glyph that is in the active glyph window will appear in the metrics window.

The metrics window has a main editing field, scroll bars, and command area with a local toolbar:

To choose a string of characters to preview or modify use the string selection control:



The popup button to the right of the control allows you to choose a pre-defined string. To the right of the button there is an options button. Click it to get access to the Metrics window's **Options** dialog box where you can customize the window.

Metrics Window Toolbar

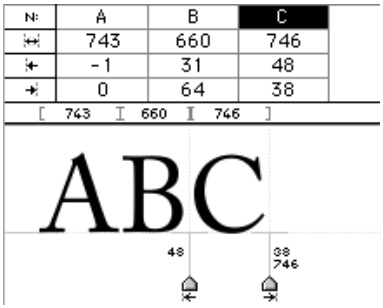
The buttons on the toolbar mean:

	Metrics mode	Switches the window to Metrics mode
	Kerning mode	Switches the window to Kerning mode
	Open metrics file	With the metrics window you can import metrics and kerning information from AFM or PFM
	Save metrics file	You can save AFM and PFM files to use them later with other fonts
	Auto	Automatically calculates metrics or kerning
	Reset kerning	In Kerning mode may partially or completely remove information about pair kerning.

Metrics Modes

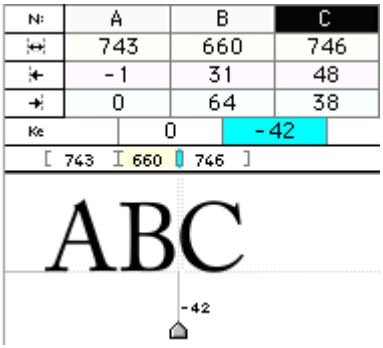
The metrics window works in three modes: *metrics, kerning* and *preview*.

In **Metrics mode** you can change the glyph sidebearings using either visual or digital controls:



In Metrics mode the string of glyphs is previewed without kerning.

In **Kerning mode** you can change pair kerning:



In **preview mode** (M and K buttons are not pressed) you cannot change glyph metrics or kerning, but you can safely preview the font in high-resolution mode.

Panels

Some TypeTool operations are accessible through Panels – small windows that are located in front of the main Font, Glyph and Metrics windows.

Use the **Window** menu to open panels.

Below is the list of all the panels available in the **Window** menu. They are described in full detail in the sections that are related to their functions, so this is only a short reference:

Transformation	Panel for digital outline transformations
Font Map	A picture representation of big Unicode fonts

All the panels can stick to either side of the screen and to each other. **To make a panel stick** just drag it close to the screen or another panel's edge.

To prevent the panel from sticking, hold down the **CMD** key while dragging the panel's title bar.

Some panels are available only under certain conditions.

Every time you exit TypeTool the positions of all toolbars and panels are stored in preferences file, so when you run TypeTool the next time, the environment will be restored.

Editing Fonts

In this chapter we will discuss the editing of fonts. A font is a collection of glyphs with similar design and some encoding and header information. The information includes font identification names, copyright data, character encoding information and other data that is necessary for font usage.

2

Opening Fonts

With TypeTool you can create new fonts or open existing fonts for modification. When you open an existing font, however, please be sure that modifying it does not violate copyright laws: some fonts are copyrighted as software so it is not legal to change them in any font editor. Carefully read the license agreement that comes with every font.

You can find two (one serif and one sans-serif) royalty-free, non-copyrighted fonts that you can use as a basis for your own fonts or characters in the TypeTool/Samples folder on your CD or on our website (www.fontlab.com).

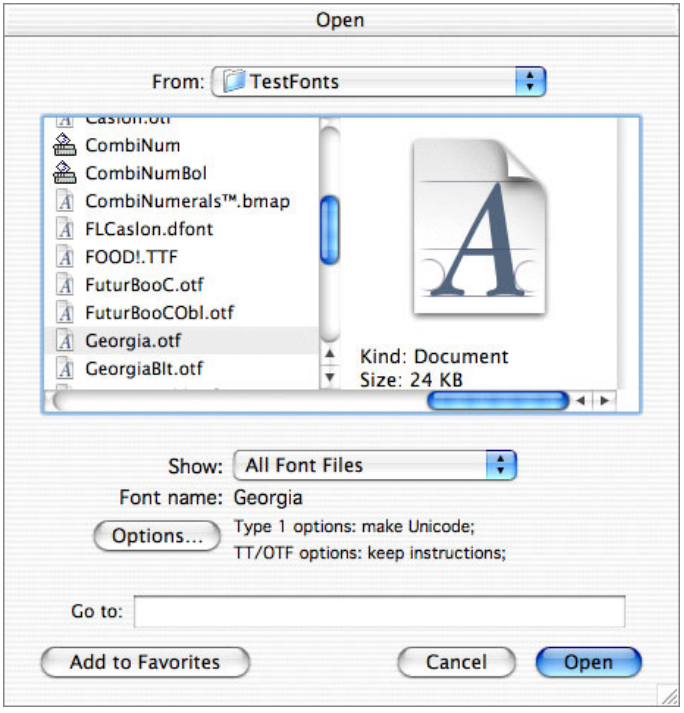
Please, note that with TypeTool you cannot edit Multiple Master and OpenType fonts. If you need to work with these fonts, we recommend you try our FontLab product:

http://www.fontlab.com/html/fontlab.html

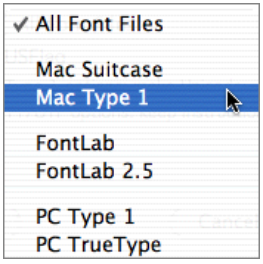
To open a font for editing, select the **File>Open** command, or

click the button on the Standard toolbar.

You will see the standard **Open File** dialog box in which you can select a font file to open. Note that in this dialog box you will see all the fonts that can be imported: Mac Suitcase (without extension or .dfont), Mac Type 1, PC TrueType (TTF), PC Type 1 in binary (PFB) and ASCII form (PFA) and fonts in FontLab formats (VFA and VFB).



If you want to list only fonts in a particular format, select that format in the popup menu located in the center of the **Open** dialog box:



When you select a font file in the files area, you will see the font name in the bottom of the dialog box.

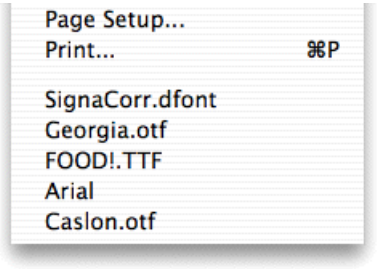
You can open multiple fonts with a single operation: just select them in the list with **Shift**-click or **Cmd**-click.

The current import options appear below the font name field. You can set the options here by clicking the **Options** button. We will discuss the import options in full in one of the next sections.

To open fonts installed in Mac OS you are better off copying the font files into some other directory and then opening them in TypeTool from that directory. Or you can simply drag-drop the font files from the Finder window into TypeTool's icon.

Most Recently Used Fonts

All fonts that you recently opened in TypeTool are added to the bottom of the **File** menu:



Next time you want them, just select the font file in the **File** menu and TypeTool will open it.

Opening Fonts with Drag-Drop

The easiest way to open fonts in TypeTool is to drag-drop font files from Finder. Even if TypeTool is not running, you can drag-drop files onto its icon on the desktop or in the Dock to run TypeTool with those fonts open.

Font Formats

TypeTool can import outline fonts in any of following formats:

Format	Extension	Description
Mac Suitcase	none or dfont	TrueType fonts in Macintosh-compatible resource-based form or Mac OS X-compatible data fork-based form
Mac Type 1	none	Single Master or Multiple Master* PostScript fonts in Macintosh-compatible resource-based form
PC Type 1 Binary	PFB	PostScript fonts in Windows-compatible binary form
PC Type 1 ASCII	PFA	PostScript fonts printer downloadable form (this format is also used in most Unix-based operating systems)
PC Multiple Master*	PFB or PFA	Special version of PostScript fonts — in binary or ASCII form
PC TrueType	TTF	TrueType fonts that are standard in Windows and Macintosh operating systems now
FontLab 2.5	VFA	Fonts in FontLab 2.5 internal format
FontLab 3.0	VFB	Native format of FontLab 3.0, 4.0 and 4.5 and TypeTool 1.0 and 2.0

*TypeTool cannot open and edit Multiple Master fonts but you may import a single master instance.

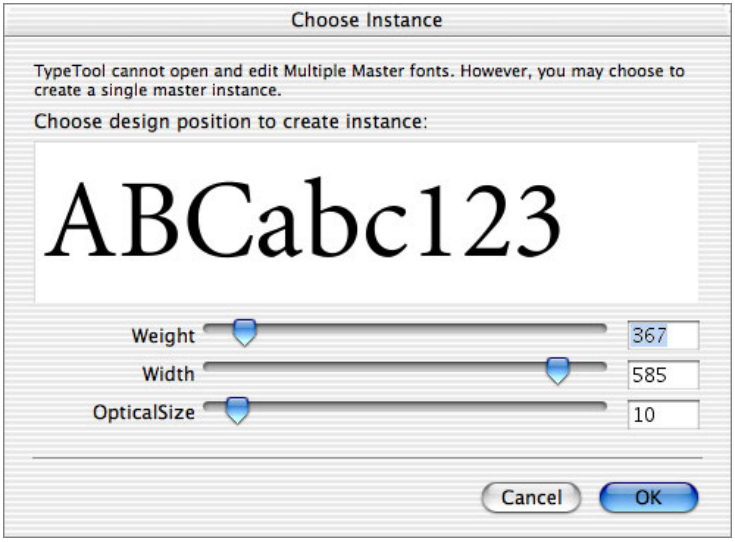
Note that TypeTool 2.0 cannot open font files that were created in FontLab 2.0. The default extension of those files is VFA (same as FontLab 2.5 fonts), but they are not compatible with TypeTool 2.0. If you want to move fonts from the FontLab 2.0 format to TypeTool 2.0, use PC Type 1 Binary as an intermediate format.

If you have fonts in a different font editor's own font format and you want to open these fonts in TypeTool 2.0, export the fonts in the Type 1 Binary format using the other font editor and then open the Type 1 fonts in TypeTool.

Multiple Master Fonts

Multiple Master fonts contain several font styles, called *master fonts*, in one font file. The Multiple Master font format is an extension of the Type 1 font format. A program that uses a Multiple Master font can select not only one of the master fonts, but also any intermediate style created by interpolation of the master fonts. So it can use not only Bold, Normal, Narrow or Wide styles, but any style in between, such as 30% Weight and 47% Width.

With TypeTool you cannot open and edit Multiple Master fonts. Instead you have the possibility to create a regular single master font from an instance of a Multiple Master font. When you select a Multiple Master font for import the **Choose Instance** dialog appears:

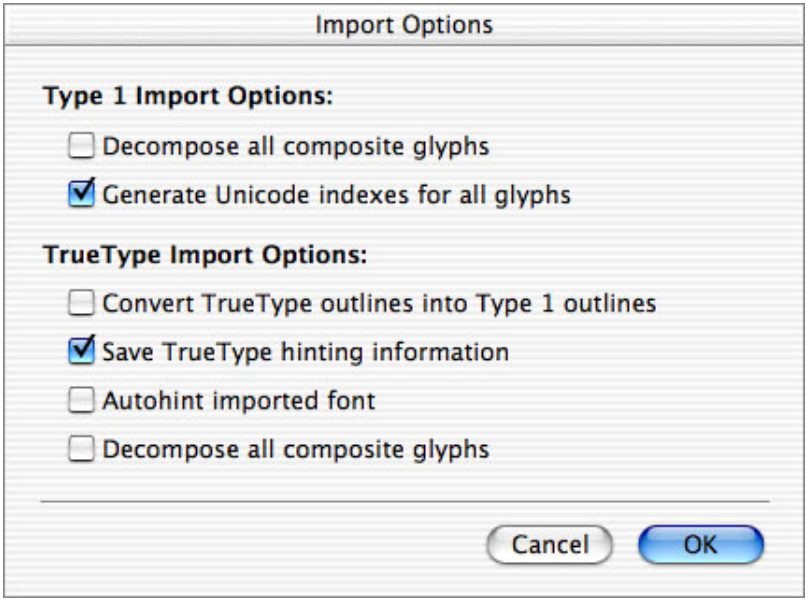


Use the controls to select the intermediate design. Click **OK** and TypeTool will generate a single master font and open it in a new Font Window.

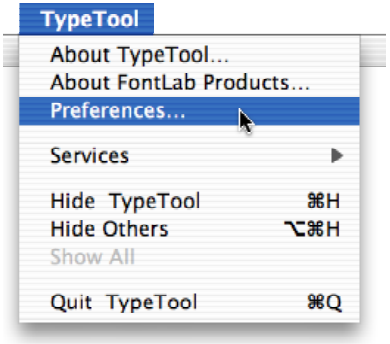
Import Options

Before opening font files you may want to set some options that can help you get started more quickly. In the following sections we will discuss all the font-opening settings. We will use some technical terms that will be discussed in more detail later, but if you want to check the definitions now you can find them in the Index.

All the opening options are accessible through the **Import Options** dialog box. **To open this dialog box** use the **Options** button in the middle-left area of the **File Open** dialog box:

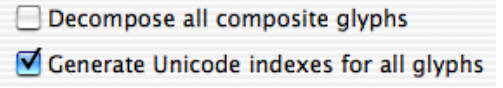


The same options are available in both **Type 1** and **TrueType** pages of the **TypeTool Preferences dialog box** accessible through the **Preferences** command of the **Application** menu:



Type 1 Import Options

There are two options in opening Type 1 fonts: **Decompose all composite glyphs** and **Generate Unicode indexes for all glyphs**.



If the first option is on then TypeTool will decompose all composite glyphs in the imported font. Composite glyphs have no unique outline themselves, but "borrow" outlines from other font glyphs. Good examples of composite glyphs are accented glyphs, like 'À', 'å' or 'ñ'. In each of these the composite character is composed of a character glyph outline and an accent glyph outline from elsewhere in the font. It's usually not necessary to decompose site glyphs on import. But if you want to markedly rearrange the glyphs and do not want to worry about composites you can use this option. Of course you can always decompose the glyphs later using TypeTool commands.

When TypeTool opens TrueType fonts it always decomposes composite glyphs that have rotated or slanted components.

The second option is usually on. We strongly recommend keeping it that way if you plan to convert your Type 1 font to the TrueType format. The TrueType font format uses Unicode indexes to access characters, so having the indexes set properly is paramount. However, if you do not plan to make a TrueType font you may switch this option off. As in the case of the first option, you can always make Unicode indexes later.

How TypeTool Makes Unicode Indexes

In the TypeTool/Mapping directory you will find the file STANDARD.NAM. This file contains a database that links the PostScript names of characters with Unicode indexes. When you import a Type 1 font and the option Generate Unicode indexes for all characters is on, TypeTool takes the name of every imported character and looks for it in the names database. If it locates the name there it takes the associated Unicode index and adds it to the character's list of indexes.

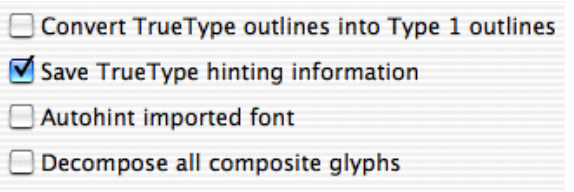
Note 1: The Names' database has about 3000 records and includes almost all known names for all European, Cyrillic, Arabic and Hebrew languages and for most symbol and dingbats fonts.

Note 2: The names' database is a text file that can be edited. You can add new records to this file at any time. Be very careful when you edit this file because incorrect records may make exported fonts unusable in some environments.

Note 3: It is possible to link more than one Unicode index to a name and more than one name to a Unicode index. If TypeTool finds several indexes linked to the name, it will assign all the indexes to the character. (Refer to the Encoding Modes section for a description of the multi-Unicode indexing method.) If more than one name is linked with the Unicode index then TypeTool will take the first one it finds as the name for the known index.

TrueType Import Options

The **Import Options** dialog box that appears if you click the **Options** button has the basic set of TrueType import options:


Convert TrueType outlines into Type 1 outlines
Save TrueType hinting information
Autohint imported font
Decompose all composite glyphs

Convert TrueType outlines to Type 1 outlines

In TypeTool you can work with Type 1 (based on 3rd -order Bezier curves) or TrueType (based on 2nd-order splines) outlines. If you open a TrueType font and plan to export this font as TrueType, it's better to keep the original outlines to avoid conversion errors. But if you plan to export your font as a Type 1 font, then we recommend converting the outlines on import. You can convert the outlines in either direction at any time using TypeTool commands.

Save TrueType hinting information

Leave this option on if you want to keep the original TrueType instructions and outlines. TypeTool will keep the imported TrueType data until you change the glyph's outline. If you import a TrueType font to rearrange glyphs or to add some new glyphs we highly recommend saving the original TrueType hinting data.

Autohint imported font

To prepare an imported TrueType font for Type 1 editing and export, you may ask TypeTool to automatically generate Type 1 hints for all the glyphs. TypeTool will use special algorithm and will make hints for TrueType or Type 1 outlines depending on the conversion setting (**Convert TrueType outlines to Type 1 outlines**).

Decompose all composite glyphs

This option works exactly the same as for Type 1 fonts.

Creating a New Font

If you want to create a new font from scratch, you select the **New** command from the **File** menu. TypeTool will create an empty font that will not have any characters and will open an empty Font Window.

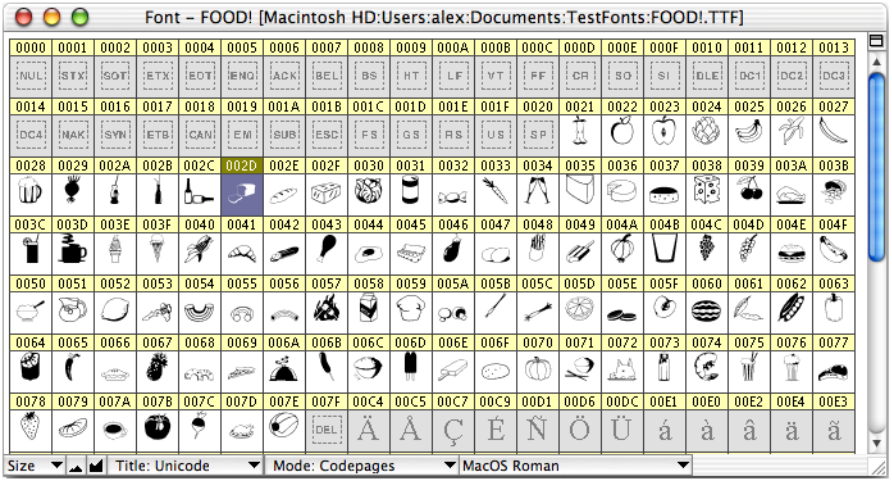
You may then add new glyphs. To make your font usable you must also fill in the important Font info fields. (See the “Font Header” chapter.)

Font Window

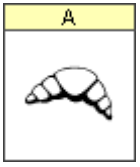
The Font Window is used to display an entire font. It opens automatically when you open an existing font for editing or choose to create a new font.

In TypeTool you can open many fonts at once and every font will have its own Font Window. The Font Window is a representation of the font, so when you close this window the font will also close.

You can do a lot of things using the Font Window — from browsing a font for a desired character to rearranging and remapping the font to editing the Font info fields. The following sections of this chapter will tell you how to use this window.

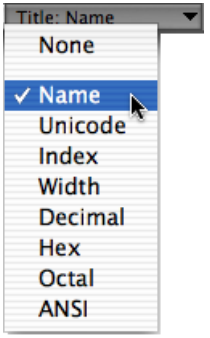


The Font Window consists of the *header*, which includes a few buttons and two popup menus (it is hidden by default), a *footer* and a *table of characters* (*character chart*), where each character is represented by a single cell:



Each cell has a *caption* showing the name of the character or its code in various forms.

The **Title popup menu** located on the Font Window footer area:

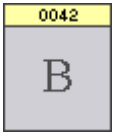


lets you **select one of the caption modes**:

None	The caption is not shown
Name	The Character name will appear in the cell caption
Unicode	The Character's Unicode index
Index	The Glyph's index
Width	The Character's width
Decimal	The Decimal character code
Hex	The Character code as hexadecimal number
Octal	The Character code as octal number
ANSI	The ANSI character that corresponds to the character code.

The characters' cells may have a gray or white background and a gray or yellow caption.

A gray cell means that there is no character defined for this place in the font. I.e. the character does not exist in this font. Instead of the character, a sample template character from the TypeTool template font is shown in the empty cell.



A yellow frame and caption in a character's cell means that this character is in the currently selected encoding or codepage (see next section). Characters that are not in the current encoding or codepage have a gray caption and no yellow frame.

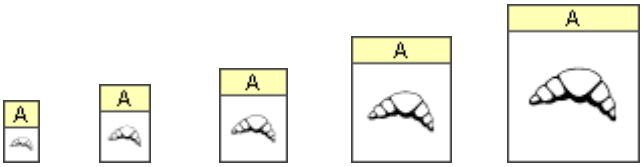
The small blue mark that appear in the corner of the character cell mean that character has more than one Unicode index assigned.

Font Window Footer

At the bottom of the Font Window you see a footer with four popup menus and two buttons:

Use the left three controls to customize the font window view. Select the desired size of glyph cell in the **Size** popup menu. Possible sizes vary from 16x16 up to 64x64 pixels. Smaller cells occupy less space but hide details.

A sample of the different cell sizes:



16x16 24x24 32x32 48x48 64x64

Use the **Zoom In** and **Zoom Out** buttons to enlarge or reduce the cells' size.

The **Title** popup menu is used to change the cells' caption contents.

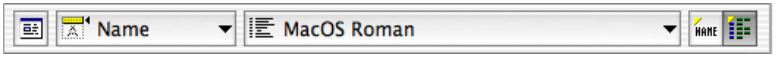
The **Mode** popup menu allows you to select one of the encoding modes: **Names** or **Codepages** mode.

And the fourth popup menu is the **Encoding** menu, which contains a list of encodings in the Names mode or a list of codepages in the Codepages mode.

Font Window Header

The Font window header is hidden by default but it can be switched on. To do this just click on the small button at the top of the vertical scroll bar:

You will see the header appears, leaving less space for characters. At the header of the Font Window you see one button on the left and two buttons in the right area:



The left button opens the **Font Info** dialog box for the current font. This is the same as choosing the **Font Info** command in the **File** menu.

The buttons on the right allow you to select one of the encoding modes:

	Switch to the Names mode
	Switch to the Codepages mode

This is the same as using the **Mode** popup menu at the footer.

And the second popup menu is the **Encoding** menu.

The Font window header can be switched on and off. This doesn't affect the font but is for your convenience only.

Encoding Modes

Support for different character indexing methods is one of the key TypeTool features.

Here's how it works:

A font is just a big collection of glyphs that are used to represent many characters (more about that in the following section). On an average screen the Font Window can show just a few hundred character cells, so we need to have some method to browse the font "through" the Font Window.

In TypeTool you can choose one of two so-called Encoding modes that allow you to select a subset of the glyph collection and show it in the top part of the Font Window for easier access.

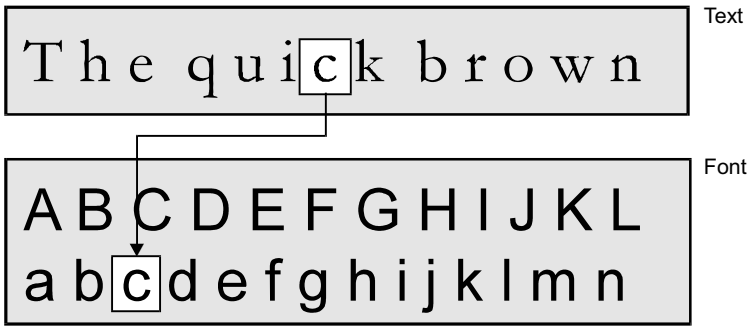
In the following sections you will find more information about encoding modes, Unicode and name-based identification and the character-glyph model.

Characters, Codes and Glyphs

A font is a collection of glyphs that usually have a common design. Each font has a header that stores top-level information, such as the font name and style, in an organised way.

Character Mapping

Text in digital form is a collection of *codes* — integer numbers. When text is entered into a computer we (with the help of a text processor program) replace characters with codes. When text is printed or previewed on the screen, the printing program performs the reverse process — replacing codes with character shapes (glyphs) that it gets from the font.



A character *mapping standard* is a set of rules that defines the relation between characters and the codes that are used to represent these characters in the computer. There are many mapping standards used in the world to help use different languages. The main difference between mapping standards is the size of the code. There are one-byte, double-byte and multi-byte mapping standards. With a one-byte mapping standard only 8 bits of information is used to represent each character, so it's possible to use only 256 characters.

A Double-byte mapping standard uses two bytes (16 bits) for every character. So it's possible to map 65536 characters. Multibyte mapping standards use from one to four bytes for every character — expanding the code space to billions of characters.

One-byte encoding standards are enough to represent most Western languages, but they run out of room if more than one language needs to be supported, or in the case of Eastern languages that use very big alphabets. Even double-byte encoding standards cannot cover all the characters of the world. The most well known and widely used double-byte encoding standard is *Unicode*.

Character and Glyph Identification

To simplify access to the characters in the font each character has an *identifier* — an integer number or a text name. All font formats allow you to quickly locate a character whose identifier is known. A printing or previewing program that wants to output text converts character codes to character identifiers in the font. This process is usually performed using a special index table, called the encoding table, which defines the relationship between character codes and identifiers.

Note that the term *character* covers *all* possible appearances of the character in *all* fonts. All these "pictures": A A A A A A A A are the same character 'A' — the first letter of the English alphabet. "Instances" of characters that have a unique design are called *glyphs*. But note that the same picture 'A' may mean different characters – the first letter of the English alphabet, the first letter of the Russian alphabet and the first letter of the Greek alphabet can be the same glyph, but are not the same character. Precise character identification methods are really necessary.

There is a difference between character and glyph identification. When we need to identify a character it means that we are identifying all glyphs that may be used to represent that character. If we are identifying a glyph, it is unique.

Characters are identified by integer number. To support exchange of information this number must be assigned according to some standard. In TypeTool all characters are identified according to the Unicode standard.

A glyph is identified by its name or by its index – its sequential number in the collection of glyphs in the font. Since the index may vary (for instance, if we remove the glyph with index 5 the indexes of all glyphs that have larger numbers, starting from 6, will change) it is not very convenient, so in most cases glyphs are identified by names.

It is very important to understand that there is no one-to-one relation between characters and glyphs. It is possible to have several characters (and character identifiers) refer to a single named glyph. On the other hand, some glyphs in the font may not be characters and may not have a character identifier.

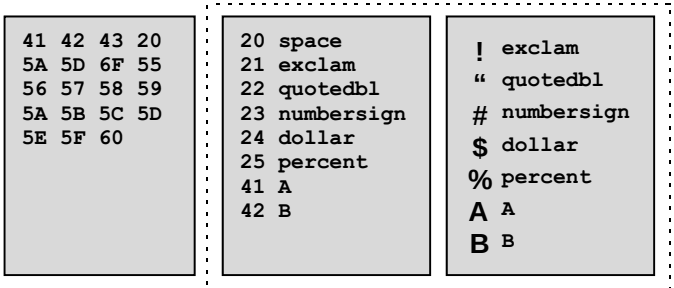
Character Identification in TypeTool

In TypeTool there are two modes of character and glyph identification:

1. **Names mode.** This mode historically is used in Type 1 fonts. An interesting point is that Names mode may have two different applications: it may be used to reference **glyphs** or **characters**. In the latter case all referenced characters must have a one-to-one relationship between character and glyph, so there is no difference between referencing the character and the glyph. Tables that are used to reference glyphs or characters in Names mode are called Encoding Tables.
2. **Codepages mode.** This is a **character**-identification method – a special table of Unicode indexes is used to map a subset of characters to the top part of the Font Window table. This table, called a codepage, may use one or two bytes to represent a character mapping record. Two-byte codepages are used to reference characters in Far-East fonts: Chinese, Japanese, Korean or Traditional Vietnamese.

Encoding Tables

The Encoding Table defines links between character codes and names, assuming that all characters in the font have name-based identifiers. The table is loaded and stays stored in memory while the font is used. Character selection processing consists of two stages — conversion of the code to a character identifier and searching the font for the location of the character description with the known identifier:



The Encoding Table can also add more flexibility to fonts. You can put many more characters into one font, assign a unique name to each character, and supply several encoding tables, allowing you to select different sets of characters in the font when you use different encodings.

For example, in symbol fonts the Greek characters take places that are usually occupied by Latin characters. With the encoding tables you can include both sets of characters. Just assign the correct names (like alpha for the ‘A’ character and A for the ‘A’ character) and later you can choose the symbol encoding to work with the Greek version of your font or choose Roman encoding to use the Latin characters.

In TypeTool you can include up to 65,000 characters in a font. You can also select any of the predefined encodings to examine a font and then include it as the default encoding upon export.

You even can create your own custom encodings and use them to properly assign names to your characters.

Names Mode

To switch the Font window to the Names mode, select **Names** in the **Mode** popup menu in the Font window footer or click the button on the Font window toolbar.

To select the current encoding for the font, use the **Encoding** popup menu. When you open it you will see the following available encodings:

Imported	The encoding that was loaded with an imported font.
Default	The default encoding for all the applications in the TypeTool family. This provides the best support for Windows or Mac OS, depending on the version of TypeTool.
Adobe Standard	Adobe Standard Encoding. This is useful if you want to create a PostScript font.
ISO Latin 1	Adobe ISO Latin 1 standard encoding.
Macintosh	Standard encoding for Macintosh computers.
Adobe Cyrillic	Standard encoding for supporting Cyrillic characters on PostScript printers and in Windows.
Macintosh Cyrillic	Standard encoding for supporting Cyrillic characters on Macintosh computers.
Symbol	Standard encoding for supporting fonts that include mathematical and other characters.
ANSI	Standard encoding used in Windows.

Additional Encodings

All encodings are stored in text files that can be edited in any text processor. This is not recommended but may help if you find that our standard encodings do not work for you.

If you want **to define a non-standard encoding**, save the file in a TypeTool/Encoding directory:

How to Create a Custom Encoding File

To create a custom encoding file:

1. Duplicate one of the .enc files located in the **Encoding** directory to use as the basis for your new encoding file.
2. Open the copy in any text editor (SimpleText or TextEdit will do) and then edit it, following the same structure that you find in the original file.
3. Change the name of the encoding and the encoding index in the first line of the file. The first line should have the following structure:

   %%FONTLAB ENCODING: 7; Adobe Symbol Encoding

   "%%FONTLAB ENCODING: " is the prefix of the file used to detect properly made encoding files and must not be changed. Note the space between ':' and the encoding index.

   '**7**' is the index of the encoding vector. You must not change the encoding vector indexes of any of the encoding vectors or they will become unusable. If you make your own encodings the indexes of your files should not be used in any of the other files. The actual value of the index is not important, so you can assign indexes like 1001 or 10001.

   The last part of the first line, "**Adobe Symbol Encoding**", is the name of the encoding vector. It starts at the first non-space character after ';'.
4. Save this encoding file with a different file name but be sure to use the .enc file extension. Put this file into the directory where all the other .enc files are located.

The new encoding will appear in the **Encoding selection** popup menu the next time TypeTool is used.

When you change the encoding you will see that the characters in the Font Window are rearranged. Some characters will move below the "yellow" zone. Remember that only characters that are in yellow cells are covered by the currently selected encoding vector.

The Unicode Standard

The *Unicode Worldwide Character Standard* (that's the full name of the Unicode standard) is a character coding system designed to support the interchange, processing, and display of the written texts of the diverse languages of the modern world. In addition it supports classical and historical texts of many written languages.

In the Unicode standard a character is identified by a double-byte index. The standard potentially can cover 65535 characters in a "basic plane" and much more using plane switching. In the current version of the standard (3.2) several hundred thousand indexes are defined. That covers almost all currently used languages, some historical languages and many pictorial characters.

TypeTool can only work with Unicode indexes from the basic plane (which potentially can cover 65,535 indexes).

The Unicode standard is used in TrueType fonts as the main character identification method. In principle TrueType fonts may be encoded with other standards, but in Windows Unicode is always used.

Until version 8.5 the Mac OS had very limited support for Unicode. Only the first 256 characters in the font could be encoded – which made Unicode unusable. Since version 8.5 the Mac OS fully supports Unicode fonts in applications that need this support.

The Macintosh version of TypeTool has special TrueType export option that makes possible the creation of Mac fonts that will work on both old and new operating systems. We will talk about this later, when we discuss TrueType exporting options, which are slightly different on Windows and Macintosh.

We recommend visiting the Unicode Consortium official Web site at:

http://www.unicode.org

to get more information about this standard.

Codepages

Codepages are tables that map character codes (one byte long) to the Unicode indexes. Depending on the size of the page, these tables may have 256 or 65,536 records, one for each possible character code. Long codepages are called double-byte codepages and are primarily used to represent codes used in Chinese, Japanese, Korean or Vietnamese languages.

Codepages are necessary because we need to somehow encode text written in different languages in the one-byte code space. So when we have a text file encoded according to some codepage, we use the codepage table to find which characters were used in this text. We may have two different texts with the same code 192 (decimal), but in one case it may mean the Russian 'А' and in the other case it may mean 'À' (Agrave).

Codepages are used not only to identify characters, but also to simplify text sorting, conversion of lowercase to uppercase characters, spell-checking and in many other applications where it is necessary to know which characters are used in the text.

Because the Unicode character identification standard covers most languages it is usually used as the destination information in the codepage tables. Here is an example of fragments from two different codepages that map the same codes to different Unicode indexes:

MacOS Roman	MacOS Cyrillic
0xC0 0x00BF	0xC0 0x0458
0xC1 0x00A1	0xC1 0x0405
0xC2 0x00AC	0xC2 0x00AC
0xC3 0x221A	0xC3 0x221A
0xC4 0x0192	0xC4 0x0192
0xC5 0x2248	0xC5 0x2248
0xC6 0x2206	0xC6 0x2206
0xC7 0x00AB	0xC7 0x00AB
0xC8 0x00BB	0xC8 0x00BB
0xC9 0x2026	0xC9 0x2026
0xCA 0x00A0	0xCA 0x00A0
0xCB 0x00C0	0xCB 0x040B
0xCC 0x00C3	0xCC 0x045B

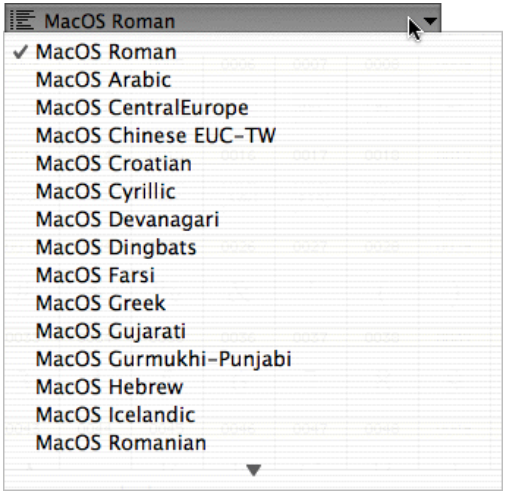
Many different codepages have been defined for many languages and different operating systems. TypeTool 2.0 includes descriptions for 124 codepages — all the known Mac OS, Windows, OS/2, MS DOS codepages plus a few others like the Polytonal Greek, Russian KOI-8 and NeXT Step codepages.

In TypeTool a codepage is a filter through which you can "look" at your font to see how it will work in different environments. For example, you might include many Unicode characters in your font and see how it would work if it was installed in OS/2 with the Arabic language selected. This gives you the opportunity to easily create fonts that will be properly encoded and will always work correctly.

To select a codepage in the Font Window:

1. Switch the Font Window to the Codepages mode by selecting **Codepages** in the **Mode** popup menu in the Font window footer or by clicking the Codepages button in the header.

2. The encoding selection popup menu will show the names of all available codepages:



   Mac OS codepages come first, MS Windows codepages follow. All other codepages are sorted according to their names.

3. Select the codepage that you want from the list and you will see the Font Window change. All the characters that are in the codepage appear in the "yellow" area. All other characters are in the "gray" area below. Select the MS Windows 1252 Latin 1 codepage and you will see how your font will look in the Windows standard (Western) codepage.

All codepages in TypeTool are defined in editable text files, so you can change any codepage if you think it is wrong (please let us know!) or you can define your own codepage. We do not recommend changing any of the codepages supplied with TypeTool. They are extensively tested and are based on the documents from the companies who supply them.

Double-byte

If your font contains many characters from one of the Far-East languages you may need to use double-byte codepages. If you select one of these codepages, you will see an additional popup menu to the right of the codepage selection menu:

This control allows you to select a “page” of the codepage. Theoretically, we could have 256 pages of 256 codes each, which would give us 65,636 codes. In practice none of the known codepages has that many codes and usually less than half of that number.

How to Make a Codepage Definition File

Codepage definition files (extension CPG) are text files that have the following structure:

%%FONTLAB CODEPAGE: 0xffff; MacOS Cyrillic

%% source: MacOS 8.0b5
0x00 0x0000

0x01 0x0001

0x02 0x0002

0x03 0x0003

The first line of this file is an identification line that is used to set the codepage name and tell TypeTool that this file is a properly composed codepage definition file. This line must be started by the text:

%%FONTLAB CODEPAGE: 0xFFFF;

The name of the codepage follows.

All other strings starting with ‘%’ are comments and are not interpreted by TypeTool.

The following strings are formed as pairs of two integer numbers in decimal or hex (starting with “0x”) form. The first number is the code of the character and should be in the 0-255 range. The second number is the Unicode index of the character and should be in the 0-65535 (0-FFFFh) range. The special Unicode index 0xFFFF is used to define codes that are not mapped to any character.

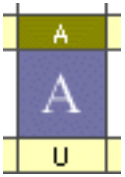
Using the Font Window

The glyph chart in the Font window is a visual representation of all the glyphs in the font. To modify the font you have to learn how to use the glyph chart: navigate, select glyphs and select commands.

From here on we will reference all characters as glyphs. We would like to clarify once more: a font is a collection of *glyphs*. Text is a collection of *codes*. An alphabet is collection of *characters*. Since we are talking of fonts, we will be using the term “glyph”.

Navigating

One of the glyphs in the Font window is the “current” glyph. It is specially highlighted:



To view different parts of the font in the Font window you can either use the vertical scroll bar or the auto-scroll mode: if you place the mouse anywhere in the chart; press the mouse button; and move the mouse cursor above the top or bottom of the chart it will scroll up or down accordingly.

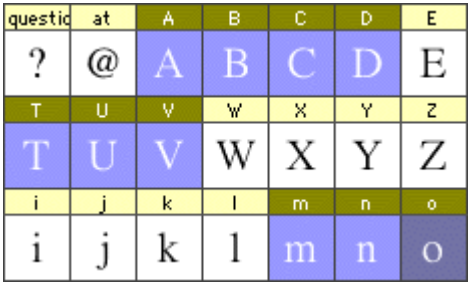
You can also use the spacebar to scroll the Font window. Press the spacebar and drag the mouse to scroll the window vertically.

Alternatively you can use the keyboard keys to navigate in the font chart:

Arrow keys	Moves the current glyph highlight one cell right, left, up or down, according to the key used
Option+Right arrow	Moves 2 cells right
Option+Left arrow	Moves 2 cells left
Page Up and Page Down	Moves the glyph highlight one screen up or down
Home	Moves the glyph highlight to the leftmost glyph on the current row
End	Moves the glyph highlight to the rightmost glyph on the current row
Cmd+Home	Moves the glyph highlight to the first glyph on the chart
Cmd+End	Moves the glyph highlight to the last glyph on the chart

Selecting

In addition to the current glyph you can select sets of glyphs in the font chart. These selections behave similarly to selected text in a text editor – you can copy selected glyphs to another place in the font or to a different font; you can apply different effects to the selection; etc. Selected glyphs have inverted colors:



To select one or more cells, press the mouse button on the first or last cell of your selection and drag the cursor across the cells you want to select. You will see the selection highlighted. If you drag the cursor outside the visible part of the chart, it will scroll accordingly. **To cancel your selection**, click on any glyph cell.

Alternative: Using the navigating keys on the keyboard, set the current cell highlight on the first (or last) cell of a selection, then press the **SHIFT** key. Move the current cell highlight (as described earlier) to select the cells.

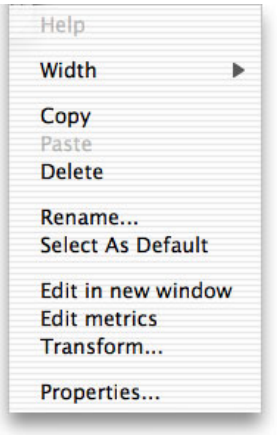
Selection does not have to be continuous. If you press the **CMD+SHIFT** keys, you can select cells in any order and combination.

Context Menu

Most commands available in the Font window can be selected from the popup menu.

To open the popup menu, CTRL-click mouse anywhere in the chart area.

Here is a sample of the Font window popup menu:



Here is what the commands mean:

Width	Allows you to easily select one of the predefined widths of the Font window.
Copy	Copies the selected glyphs onto the Clipboard. Same as the **Copy** command from the **Edit** menu.
Paste	Places glyphs from the Clipboard into the font starting from the first selected cell. Same as the **Paste** command from the **Edit** menu.
Delete	Deletes the selected glyphs. Same as the **Delete** command from the **Edit** menu.
Rename	Opens a rename glyph dialog box.
Select as default	Selects the current glyph as the "default glyph" that is used in Type 1 fonts to represent glyphs that are not present in the font.
Edit in new window	Creates a new Glyph window and opens the current glyph in it.
Edit metrics	Opens the Metrics window with the currently selected glyphs.
Transform	Opens the transformation dialog box. Refer to the "Transformations" chapter for more detailed information about available transformations.
Properties	Opens the glyph properties panel for the current glyph or selected glyphs.

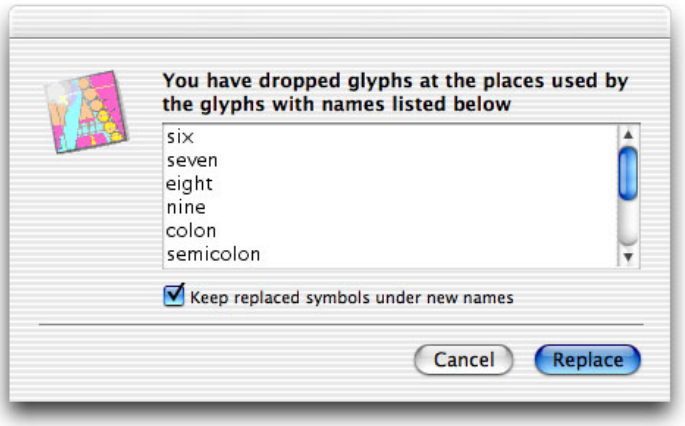
Moving Glyphs

You can change the positions of glyphs in the font chart just by moving them to a new place.

To move glyphs in the font chart:

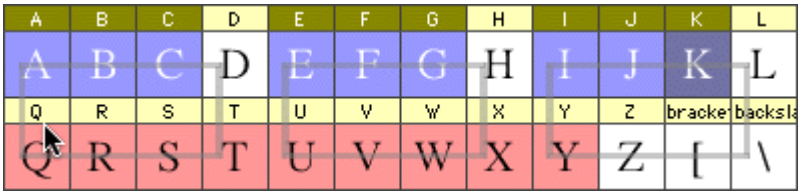
1. Select the glyphs that you want to move.
2. Position the mouse cursor on the selected glyphs.
3. Press the mouse button.
4. Drag the glyphs to the new position. Release the button to finish moving or press the **Esc** key to abort.

If you move glyphs over the cells of existing glyphs, you will see a dialog box prompting you to choose whether to replace the existing glyphs or save them by moving them to the end of the encoding:



Leave **Keep replaced symbols under new names** checked to save the glyphs (I.e. put the new glyphs in the cells and move the existing glyphs to cells at the end of the encoding) or clear it to replace them (I.e. delete the existing glyphs).

Note that even if source selection is not continuous the destination selection *will* be continuous:



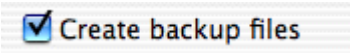
Saving the Font

Most of the font-modification operations are not undoable, so we recommend you save your work regularly.

To save a font that you have opened from an existing font file (in TypeTool or other format), use the **File>Save** command or click on the **Save** button on the Standard toolbar.

To save all opened fonts click the **File>Save all** command or this button on the Standard toolbar: .

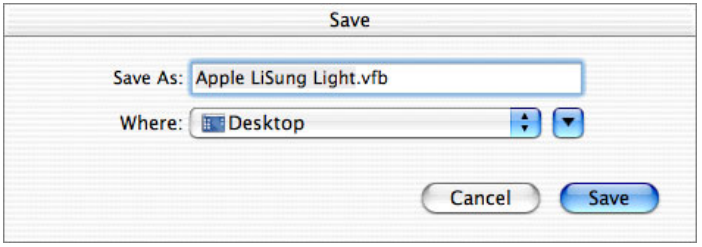
Font(s) will be saved in FontLab format (vfb extension) to the directory where the original font was opened (imported). If this option in the **General** page of the **Preferences dialog box** is active:



TypeTool will create a backup copy of the existing font file before overwriting it with the new one.

If you are working with a new font or you want to select the destination directory or change the name of the file, use the **File>Save As...** command.

After you select **File>Save As...** in the menu, you will see the standard **File Save** dialog box:



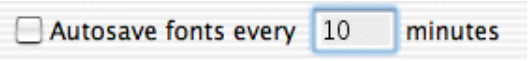
Choose the destination folder, enter the file name and click **Save** to save your font.

Autosave

If you want to protect yourself from system or program crashes you can use the Autosave function that will periodically save the current font.

To activate and customize this feature, open the **Application>Preferences** dialog box and select the **General** page.

In the top part of the page, you will see the **Autosave** controls:



Use the check box to activate Autosave and enter the time interval (in minutes) at which you want to save the font.

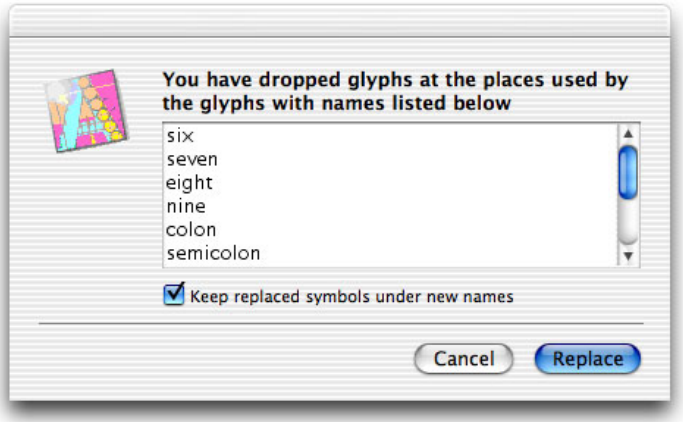
Font will be saved into the Autosave folder (subfolder in the TypeTool directory) and will be named using the following structure:

Fontfilename.save.vfb, where fontfilename is the name of the font file to which you manually saved the font the last time.

If **Autosave** was active and you have a system or program crash, you can open your last saved font from the TypeTool/Autosave folder.

Copying and Pasting Glyphs

To copy selected glyphs, select the **Copy** command from the **Edit** menu. Note that this copies not only the glyph outline, but also the glyph information, such as its name. The selected glyphs will be placed in the Clipboard and **can be pasted** into the same font or into another font by the **Paste** command from the same menu. Glyphs from the Clipboard will be placed starting from the first selected glyph in the destination font. If the destination position is occupied by existing glyphs a warning dialog box appears:



If you select the **Cut** command instead of the **Copy** command the glyphs will be copied to the Clipboard but will be deleted from the source positions.

You can choose the **Paste** command when the Glyph window is active. In this case the outlines layer of the first copied glyph will be added to the open glyph.

Copying Glyphs to Another Font

You can use two methods to copy glyphs from one font to another:

1. Use the **Copy** and **Paste** commands from the **Edit** menu as described in the “Copying glyphs” section, or
2. Drag them to the other font and drop them there. The drag-and-drop method is easier and more visual.

Copying Composite Glyphs

If you copy composite glyphs (instead of having their own outlines composite glyphs are built from references to other glyph outlines) to another font, TypeTool will try to not decompose (replace references to glyph with actual glyph copies) them. Instead it will try to find matching components in the glyph set that was copied or, if some components are not present there – in the destination font.

If TypeTool can completely restore composites in the destination font it will even keep TrueType hinting programs for these glyphs.

Drag-and-Drop of the Composite Glyphs

If you prefer to use the drag-and-drop method to copy composite glyphs you have one additional option: when you drop a composite glyph and TypeTool finds that one or more of its components were not selected to copy and do not present in the destination font, it shows a message asking you if you want to copy all the missing components. If your answer is Copy, then TypeTool will automatically append all the necessary components to the destination font so that all the composites stay unchanged.

Duplicating Unicode Indexes

In TypeTool you may assign more than one (up to 63, actually) Unicode indexes to a glyph. Visually this means that a glyph that has several Unicode indexes will appear several times when the Codepages modes is selected in the Font window. Both the original copy of the glyph and the duplicates are marked by a small blue mark in the left-top corner of the glyph cells.

To make a duplicate of a glyph, select it (you may select many glyphs at once); position the mouse cursor on the selection; press the **COMMAND** and **OPTION** keys; press the mouse button and drag the selection to the place where you want it to be duplicated. It is important to have the **COMMAND** and **OPTION** keys pressed when you release the mouse button.

You can later correct Unicode indexes and names assigned to the glyph by using the Rename Glyph dialog or the Glyph Properties panel (described later).

Deleting Glyphs

To remove glyphs from the font

1. Select the glyphs that you want to remove.
2. Select the **Delete** command from the **Edit** menu or from the popup menu. Or, press the **DELETE** or **BACKSPACE** key on the keyboard.
3. A dialog box appears asking you if you are sure that you want to delete.

✓ **Note 1:** Deleting glyphs from the font is not undoable, so save your work before deleting glyphs.

✓ **Note 2:** If you are in Unicode mode and deleting glyphs with the blue mark in the top-left corner, they will be removed without any questions because they are just one of the indexes of a multi-Unicode glyph.

Creating New Glyphs

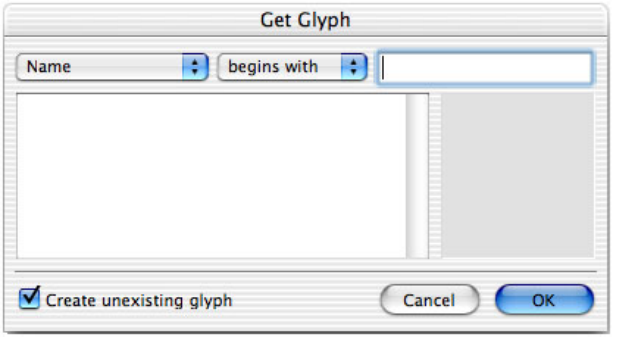
If you want to create a new glyph in an empty place in the font (a gray cell in the Font window), double-click the cell.

If you want to create a group of new glyphs with a single command, select the empty cells and use the **Reset** command in the **Glyph** menu.

If you are creating glyphs in the “yellow zone”, names and Unicode indexes are assigned to the newly created glyphs according to the selected encoding table.

Searching for Glyphs

Sometimes you need to find a particular glyph in your font, especially in large fonts. Select the **Choose Glyph** command in the **Glyph** menu or press **CMD+BACKSPACE** on the keyboard. You will see a dialog box:

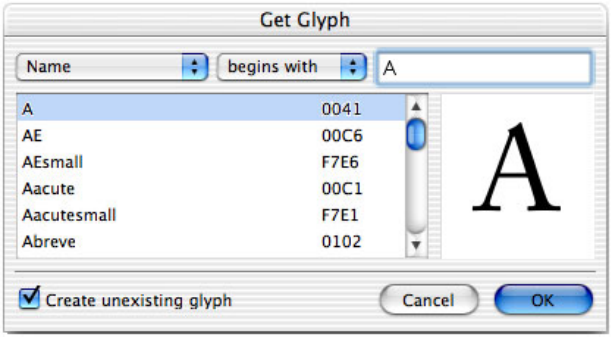


To find a glyph:

1. In the left-top popup menu select the method by which you want to search for the glyph:

Name	Searches for the glyph name
Code	Searches for the decimal code of the glyph in the current encoding or codepage
ANSI glyph	Searches for the glyph that is mapped to one of the ANSI glyphs in the selected codepage or encoding
Unicode index	Searches for glyphs with Unicode index attributes given
Width	Searches for glyphs with the width in the selected range
Bottom, Top	Searches for glyphs whose bottom or top line falls in the specified range
Components	Searches for glyphs that have the specified number of components.

2. In the popup menu to the right of the method select the comparison factor: begins with, equals to, less than or more than.
3. In the right-top editing field enter the information (depending on your selection) that will be used to find the glyph.
4. The names of all the glyphs that match the criterion will appear in the list.

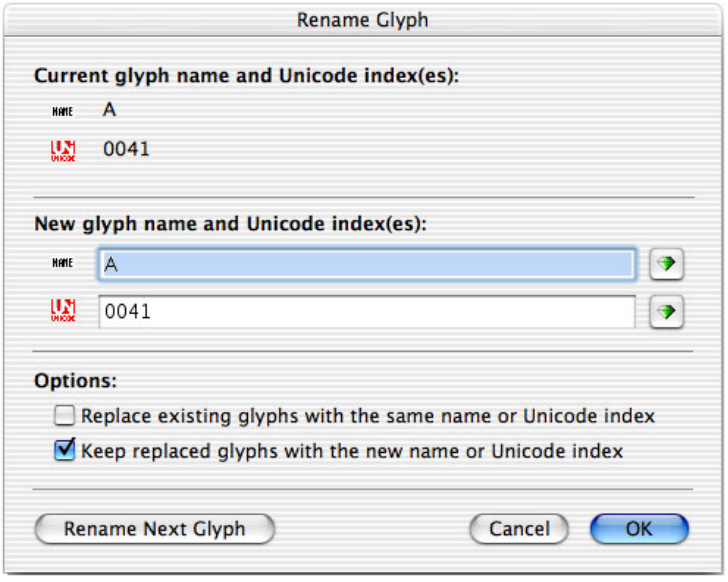


Select the glyph name that you want (its preview appears in the preview panel) and press **OK**, or enter more information to narrow your search.

Renaming Glyphs

Usually it is not necessary to manually rename glyphs because their names and Unicode indexes are assigned automatically when you move glyphs in the Font window. But if you want to see the information and correct it, select the **Rename Glyph** command from the **Glyph** menu. Or just press **CMD**+\ on the keyboard.

You will see a dialog box:



In the top part of the dialog box you see the current name and Unicode index (indexes) of the glyph. In the middle there are two edit fields where you may change the information. Below them lie the options controls.

To change a glyph's name enter a new name in the **Name** field. If this glyph has a properly assigned Unicode index and you want to find the name mapped to that index in TypeTool's database press the **Auto** button to the right of the edit field and TypeTool will fill in the **Name** field for you.

If the option **Replace existing glyphs with the same name or Unicode index** option is not checked then, if you enter a name that is already assigned to one of the font's glyphs, the **OK** button will be disabled and you will not be able to assign that name. Switch the option on to allow TypeTool to replace glyphs. Use the next option to control how TypeTool does the replacement.

Use the **Unicode** edit field to change a glyph's Unicode indexes. You may enter more than one Unicode index separated by a space. Use the **Auto** button to find the Unicode indexes mapped to a glyph's name in TypeTool's database.

Press the **OK** button to assign a new name to the glyph. You will see that the glyph moves to a new place in the Font window depending on the currently selected encoding vector or codepage.

If you want to rename more glyphs, press the **Rename Next Glyph** button. A new name will be assigned to the current glyph (as if you had pressed the **OK** button) and data from the next glyph will appear for editing.

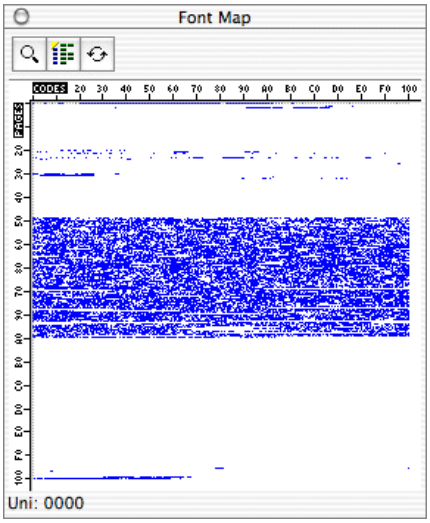
The Font Map Panel

When you work with really big Unicode-encoded fonts, you may need to have an overview of your whole font. TypeTool has a special panel, called the *Font Map*, which can represent the entire Unicode code space as a 256 x 256 picture where every pixel represents a double-byte code.

Every pixel row in this picture represents a Unicode page – 256 Unicode indexes which begin with the same code. For example, codes A700-A7FF will form one row.

Every pixel in the row represents an individual code.

To open the **Font Map** panel, use the **Font Map panel** command in the **Window** menu. You will see a panel that consists of the code picture, toolbar and status bar:



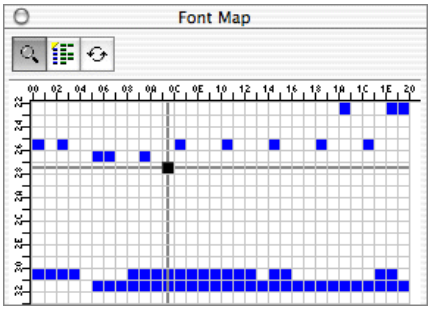
The buttons on the toolbar mean:

Turns on/off zoom mode

Changes the Font Map to double-byte codepage mode

Updates the contents of the Font Map.

By clicking the button on the toolbar you can zoom in on part of the Font Map:



In this mode it is much easier to manage individual codes. To scroll a zoomed Font Map, press the mouse button and drag the cursor beyond the Map borders.

If you click the Font Map, you will see the current Unicode index appear on the status bar below the Map picture. The current code is highlighted with a cross hair.

Double click any code in the Map **to jump to the glyph that is mapped to it**.

Font Map automatically tracks changes you make to the font. If you are not sure that it is correctly updated, click on the button **to manually update the Map**.

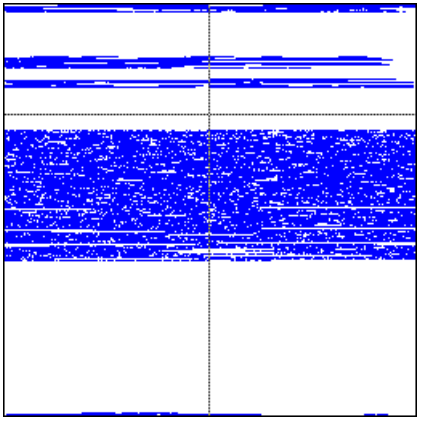
Managing Double-Byte Codepages

If you are working on a CJKV (the acronym for Chinese, Japanese, Korean and Vietnamese) font, you may want to look at your font in a double-byte codepage.

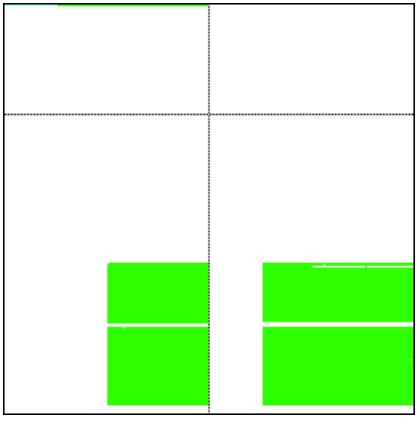
Open the Font Map panel and in the Font window of your font select one of the double-byte codepages.

You will see this button enabled in the Font Map toolbar. Click it and you will see the Font Map rearrange to represent your font with the applied double-byte codepage. In this mode every row represents 256 glyphs that are "assigned" to the specific first byte.

In the following picture you can see a Traditional Chinese font in Unicode mode (in the left picture) and in Codepage 950 mode (right):



Unicode mode



Codepage 950

In the codepage mode **green pixels** represent codes in the codepage that are covered by one of the glyphs in the font. **Cyan pixels** mean codes in the codepage that are not covered by any glyphs in the font.

Applying Actions

You can find many actions in the **Glyph** and **Tools** menus that can be applied to the open glyph in the Glyph window (see next chapter), but most of them are applicable to the glyph(s) selected in the Font window.

To apply an action, select the glyphs in the Font window and choose the appropriate command in the **Tools** or **Glyph** menu. For example, to convert glyphs from TrueType outlines to Type 1 outlines, select the glyphs that you want to convert and choose the **Tools>To Type 1 Curves** command. You will see the warning message:

Do you want to continue and transform glyphs?
You will apply selected action to the 12 glyphs. This operation is not undoable.
No
Yes

Click **Yes** and TypeTool will apply the action to all selected glyphs.

You can find descriptions of all actions in the chapter “Glyph Window”.

Exporting Fonts

You know how to open fonts, how to rearrange characters and change encodings and how to save fonts in TypeTool format to keep your changes. Now it's time to learn how to generate fonts in formats that other programs can understand.

Please note that you cannot import and export OpenType and Multiple Master fonts. If you want to work with fonts in these formats consider using our FontLab product. More information about it is available at this web page:

http://www.fontlab.com/html/fontlab.html

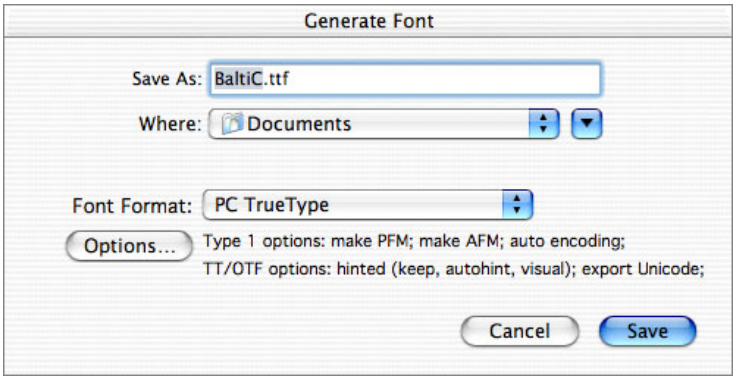
Font Formats

Macintosh version of TypeTool can work with three font formats and their variations:

PC Type 1	Sometimes fonts in this format are called PostScript fonts. Usually they consist of two or three files: a file with the .pfb extension and one or two additional files needed by the OS to get information about the font header and font metrics. You need Adobe Type Manager to use these fonts on Windows (if you are not using Windows 2000 or Windows XP).
Mac Type 1	On Macintosh computers fonts in this format are always called PostScript fonts. Usually they consist of two resource files: a suitcase file with at least one bitmap font (needed by the Mac OS to get information about the font family) and the file with 'post' resources containing font-definition data.
PC TrueType	The standard format for Windows fonts since version 3.1 and for the Mac OS. The font is stored in a single file that has a .ttf extension. Mac OS now supports PC TrueType.
Mac TrueType	The font is stored in a single resource (Pre Mac OS X) or plain data-fork (on Mac OS X with .dfont extension) file that is called a suitcase. One suitcase can contain up to 4 TrueType fonts combined in a family.
FontLab	This is TypeTool's own font format that potentially can contain a superset of information that you can export to other formats. You already know how to open and save FontLab fonts.

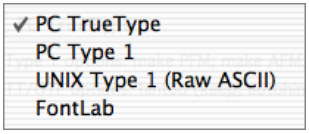
Exporting for PC

To save a font in PC Type 1 or PC TrueType format use the **File>Generate Font** command. You will see the **Generate Font** dialog box:



The top part of the dialog box is standard and there you choose a destination folder and enter a file name for the font file. By default TypeTool will choose the folder where you saved fonts last time.

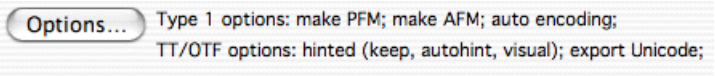
Below the standard part of the dialog box is format selection popup menu. Choose the destination font format there:



Available formats are:

PC TrueType (ttf)	TrueType font format.
PC Type 1 (pfb)	Single-master Type 1 font format in a binary form. This is the format for Type 1 fonts that is expected by Windows.
UNIX Type 1 (raw ASCII) (pfa)	Single-master Type 1 font format in ASCII (7-bit) form. This format is used in most UNIX systems and can be directly downloaded into PostScript printers.
FontLab (vfb)	TypeTool internal font format. This format it is used when you select the **File>Save As...** command.

Below the format selection popup menu is the **Options** button (that opens the **Export Options** dialog box described later in the chapter) and a legend field that describes the currently selected options for TrueType and Type 1 formats:



Check the export options in the dialog and press the **Save** button to export the font, or **Cancel** – to return back to font editing.

Exporting for Macintosh

As far as Mac OS X now supports PC TrueType fonts you may choose this format available in the Generate Font dialog box described in the previous section. We will describe here the font suitcase generating procedure.

Font Suitcases

On the Macintosh fonts that belong to a font family are physically combined into a single file, called a font suitcase. The suitcase contains information about the font family in general and refers to the records that describe the fonts' style, encoding, metrics and kerning information and some other data that is necessary for the Mac OS to use the font.

When Type 1 fonts are used on Macintosh, files that contain actual Type 1 font-definition data are stored in a separate file that is referenced from the font suitcase.

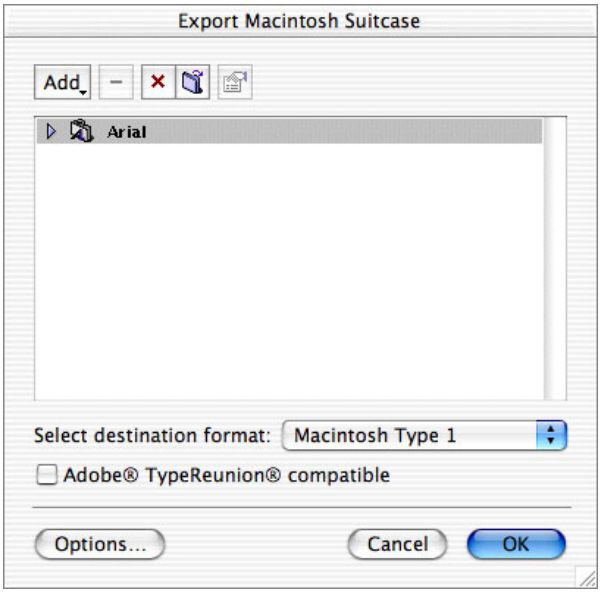
TypeTool has a special dialog to compose and export Macintosh suitcases – the **Export Macintosh Suitcase** dialog. You can group several open fonts to collect them in a new destination suitcase or to add them to an existing one.

To build a proper suitcase, one must fill in the fonts' Font Info fields properly (described in full detail in the "Font Header" chapter). There are several basic rules:

1. One suitcase can contain fonts from only one font family. This means that the **FOND Name** fields of all the included fonts must be the same.
2. TypeTool designates the Macintosh's bold style for a font if the font's **Weight** field is set to "DemiBold" or more.
3. TypeTool designates the Macintosh's italic style for a font if the **Font is italic** checkbox is set.
4. A maximum of 4 fonts can be combined as linked styles: style Plain or Normal (Mac font style nr. 0), style Bold (Mac font style nr. 1), style Italic (Mac font style nr. 2) and style BoldItalic (Mac font style nr. 3).

Building Font Suitcases

To save a font in traditional suitcase-based Mac Type 1 or TrueType format use the **File>Generate Mac Suitcase** command. You will see a special suitcase managing dialog box:



The **Export Macintosh Suitcase** dialog box consists of 4 parts: a toolbar on the top, a list in the middle, an options area and three buttons at the bottom. In the simplest case you can just press the **OK** button and get a font, but usually some management is needed.

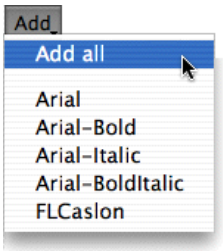
By default the list already contains a suitcase that will be generated from the font that was active when you select the **File>Generate Mac Suitcase** command. Click on the triangle on the left of the suitcase icon to see its contents. If you feel the style is not what you've expected, click **Cancel** and check the fonts' Font Info fields carefully.

The toolbar contains the following buttons:

Add	Opens the menu allowing you to add open fonts to the list
–	Deletes the selected font suitcase from the list or font from the suitcase
×	Completely clears the contents of the list
	Opens the standard File Open dialog allowing you to open suitcases from the disk
	Opens the **FOND Info** dialog box for the selected suitcase or **Font Info** dialog box for the selected font.

What you can do with these commands?

To add all fonts that have been opened with TypeTool in their Font windows click on the Add button and select the **Add all** command from the menu:



TypeTool checks the FOND Name field in every open font and creates suitcases one for every different FOND name. If several fonts have this field the same they will be combined in one suitcase with this name.

If the list of suitcases is not empty (and it is so after you select the **File>Generate Mac Suitcase** command), you have the possibility to add only those fonts that are compatible with this suitcase in the list: choose the **Add possible** command from the **Add** pulldown menu.

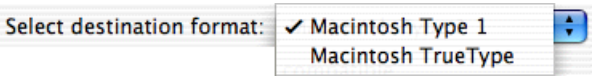
To remove all the fonts from the list click on the × button.

To remove a suitcase or a particular font in a suitcase select it in the list and click on the − button. If you remove the last font from the suitcase, then the suitcase is removed too.

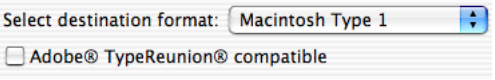
To add new fonts to existing suitcase, first clear the list with the × button, then click on the button. The standard File Open dialog box appears where you may choose the suitcase from the disk. Press the **Open** button in the dialog and the selected suitcase will appear in the list. Click on the Add button and select the font that you want to be added to the opened suitcase. If the Font Info fields of the added font are filled properly it will appear in the corresponding style of the font suitcase. You have the possibility to drag-drop individual fonts between the styles in a suitcase.

Options

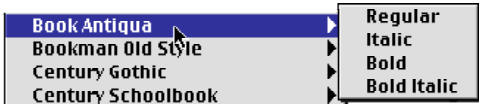
Before generating a Macintosh suitcase you have **to choose the font format**. Select Macintosh TrueType or Type 1 in the popup menu:



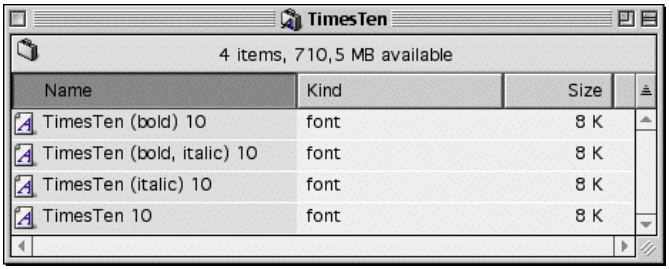
When Type 1 is selected additional option is available:



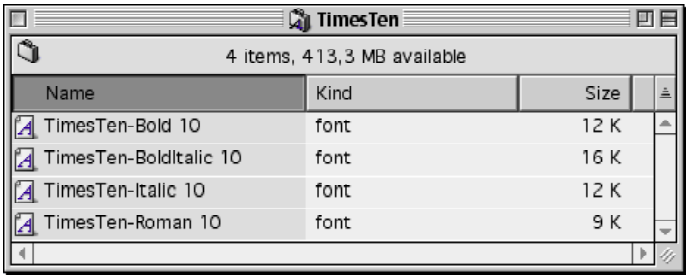
If this option is checked, TypeTool will generate a font suitcase compatible with Adobe® TypeReunion®, which lets the font's styles appear in the hierarchical menu of the font menu:



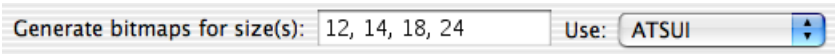
Finder's view of a suitcase created with the "Adobe® TypeReunion® compatible" checkbox switched off:



The Finder's view of the suitcase created with the "Adobe® TypeReunion® compatible" checkbox switched on is as follows:

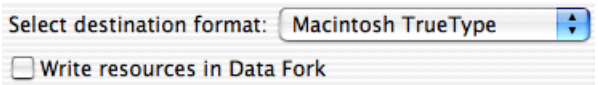


On Macintosh Type 1 fonts have to be generated with at least one accompanying bitmap font in a 'NFNT' resource. You can define the point sizes that will be generated on the Type 1 page of the **Application>Preferences** dialog box:



You may select a rasterizer for generating bitmaps. Choose among the build-in rasterizer or ATSUI (Apple Type Services for Unicode Imaging) on Mac OS X. If you are using classic system and have ATM installed, then choose among the build-in and ATM's rasterizers.

When TrueType is selected another additional option is available:



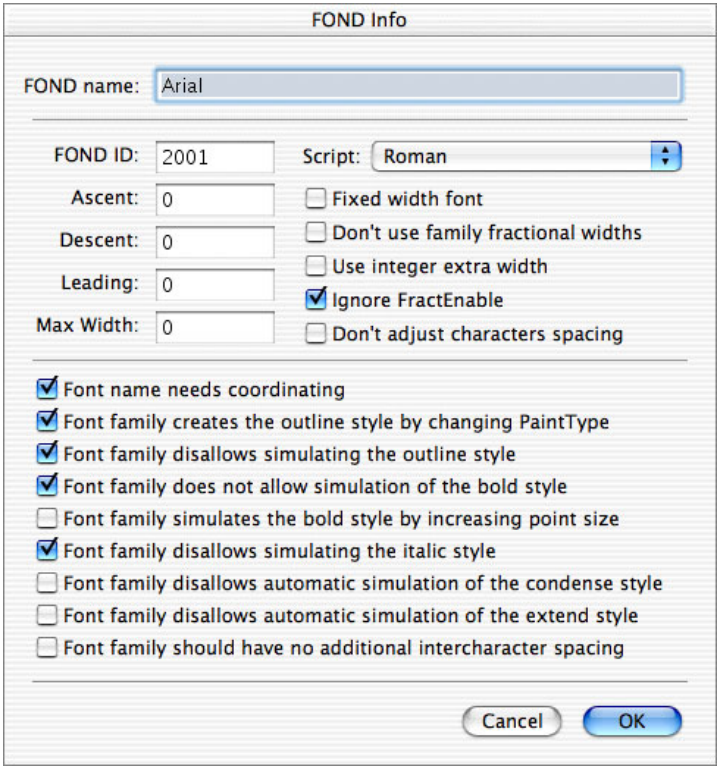
If this option is checked, TypeTool will write a font suitcase as data-fork-based file supported by Mac OS X only. Leave this option unchecked if you want your font to be compatible with Classic.

Other font exporting options are available in the **Export Options** dialog box that appears when you click on the **Options** button at the bottom of the **Generate Mac Suitcase** dialog box and described later in the chapter.

After you finished managing suitcases in the list press the **OK** button and select the destination for the fonts in the standard Save dialog. Name the suitcase file if needed and press the **Save** button to generate fonts.

Family Info

To view and edit the font family information select the suitcase in the list and click on the button. The FOND Info dialog box appears:



Besides the FOND name you may edit parameters in the following groups:

Font family properties:

FOND ID and Script	FOND resource identifier (or family ID number) lying in the range of the particular script. Changing the script in the popup menu to the right will automatically change FOND ID and vice versa
Fixed width font	If this option is switched on, the font will be treated by the Macintosh system as one with characters of fixed width (monospaced). Otherwise, the font is treated as proportional
Don't use family fractional widths	If this option is switched on, the system will not use the global family widths table
Use integer extra widths	If this option is switched on, the system will use the family style extra widths table (Family Style Property Table)
Ignore FractEnable	If this option is switched on, the system will use the family style extra widths table (Family Style Property Table) even if the option **Don't use family fractional widths** is switched off
Don't adjust characters spacing	This option represents the 11th bit of the family flags, which is usually set to zero.

Font metrics:

Ascent	The maximum height above the baseline reached by characters in this family fonts
Descent	The maximum depth below the baseline reached by characters in this family of fonts. The depth is usually a negative number
Leading	Maximum leading for the family. The leading value is usually set to zero
MaxWidth	Maximum character width for the family.

Style mapping flags (Font Class):

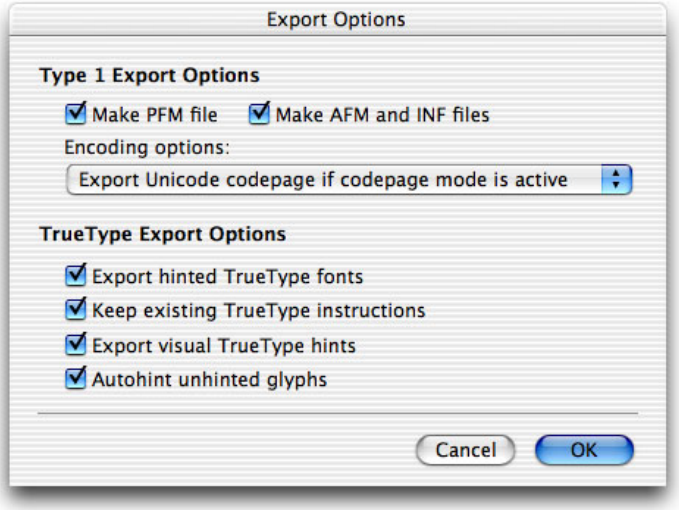
Font name needs coordinating	This option is switched on if the font name needs coordinating
Font family creates the outline style by changing PaintType	When this option is switched on, the **Outline** style of the family will be created by changing PaintType, a PostScript variable, to 2
Font family doesn't allow simulation of the outline style	This option is switched on if the font family disallows simulating the **Outline** style by smearing the glyph and whiting out the middle
Font family doesn't allow simulation of the bold style	This option is switched on if the font family disallows simulating the **Bold** style by smearing the glyphs
Font family simulates the bold style by increasing point size	This option is switched on if the font family simulates the **Bold** style by increasing the point size
Font family doesn't allow simulation of the italic style	This option is switched on if the font family disallows simulating the **Italic** style
Font family doesn't allow simulation of the condensed style	This option is switched on if the font family disallows automatic simulation of the style **Condensed**
Font family doesn't allow simulation of the extended style	This option is switched on if the font family disallows automatic simulation of the style **Extended**
Font family should have no additional intercharacter spacing	This option is switched on if the font family should have no additional spacing other than the space character.

To get full information about the parameters represented in the FOND Info dialog, refer to Inside Macintosh: Text:Font Manager:
http://developer.apple.com/techpubs/mac/Text/Text-181.html

Export Options

There are two methods of selecting export options. The basic set of TrueType and Type 1 options is available if you press the **Options** button in the **Generate Font** or **Export Mac Suitcase** dialog box. If you need to get access to all the options, use the **Type 1** and **TrueType** pages of the **Application>Preferences** dialog box.

This is what you see if you click on the **Options** button in the **Generate font** or **Export Mac Suitcase** dialog box:

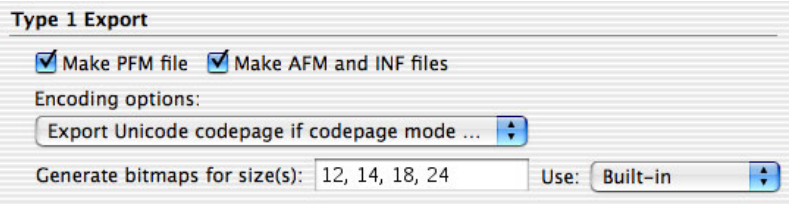


The **Export Options** dialog box consists of two areas: *Type 1 export options* and *TrueType export options*. In the following sections we will discuss exporting options in full detail.

Type 1 Export Options

Type 1 exporting options in the **Export Options** dialog box are a subset of the options accessible in the **Application>Preferences>Type 1** dialog box. Usually the subset is enough, but sometimes you may need to customize the full set.

To get access to the full set of Type 1 export options, open the **Application>Preferences** dialog and select the Type 1 page:



Make PFM File

Switch this option on to create a PFM (Printer Font Metrics) file when exporting a PC Type 1 font. PFM files are used in Windows to register Type 1 fonts. They contain metrics, kerning and, partially, font header information. You cannot install a Type 1 font on Windows without a PFM file. This option is ignored when saving Macintosh Type 1 fonts.

Make AFM and INF files

Switch this option on to make AFM (Adobe Font Metrics) and INF (font INFormation) files when exporting a Type 1 font. These files are text files and contain descriptions of the font metrics, kerning and header (font names, weight, width, encoding and other information). It is possible to install a Type 1 font on Windows with Adobe Type Manager if you don't have the PFM file but do have the AFM and INF files, because ATM will automatically build the PFM file using data from the AFM and INF files.

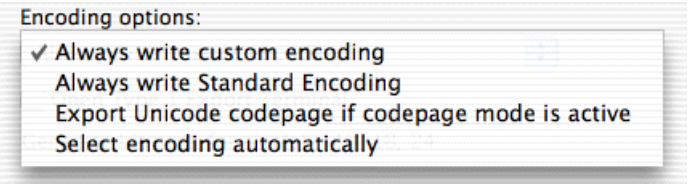
The AFM file is necessary to install a Type 1 font (in ASCII form, with ".pfa" extension) on most Unix-based operating systems.

When saving Macintosh Type 1 fonts only AFM files are created.

To install an exported Type 1 font in Windows you must have the PFM or AFM+INF files. We recommend making *all* these files when you finally produce a font so that your font will be compatible with various environments.

Encoding Selection Options

The dialog has a list of possible encoding-export options:


Encoding options:
✓ Always write custom encoding
Always write Standard Encoding
Export Unicode codepage if codepage mode is active
Select encoding automatically

When a Type 1 font is exported you have the choice of including a custom encoding vector in its body or just exporting a StandardEncoding key word that means that the font has the standard set of glyphs and that the standard system remapping tables may be applied to it.

StandardEncoding is the recommended choice if your font is a typical Latin font. If the font is in a non-standard encoding, like Cyrillic, Greek, or Arabic, you must export it with the specific encoding. In this case the encoding that is currently selected in the Font window will be saved with the font and system will use it to access glyphs. If your font contains many glyphs that can cover more than one encoding, then system will take the main encoding (if present in the font, or if assumed, as in the case of StandardEncoding) as the base encoding and will take all the other possible encodings from it.

How ATM Interprets a StandardEncoding

When a Type 1 font has StandardEncoding ATM assumes that this font includes all the glyphs from the first 128-glyph range (digits, alphabet and basic punctuation) and the European glyphs (128-255 range). The first 128 glyphs are called the "*top zone*". The 128-255 range is called the "*bottom zone*". The Adobe StandardEncoding includes very few glyphs from the bottom zone compared to the number of glyphs in the Macintosh Roman encoding. When a Type 1 font in StandardEncoding is installed with ATM, ATM uses a special encoding instead of the "real" StandardEncoding as it is documented in the Type 1 format specification. This special encoding is called the Default Encoding in TypeTool. So if you create a StandardEncoding font and want to see how it will work in Mac OS, select the Default Encoding in TypeTool.

Here is an explanation of the possible encoding export options:

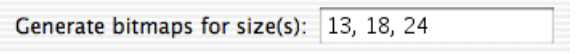
Always write custom encoding	Always saves the custom encoding with the font. Never writes StandardEncoding. The encoding currently selected in the Font window will be exported.
Always write StandardEncoding	Exports the font with the StandardEncoding if the Font window is in Names mode and MS Character Set is "Western (Latin1) CP1252 / ANSI"
Select encoding automatically	Exports the font with the StandardEncoding when the Font window shows StandardEncoding, Default Encoding or Mac Roman. Exports a custom encoding if any of the other encodings is selected or if the Font window is in Codepages mode.
Export Unicode codepage if codepage mode is active	Exports a custom encoding based on the currently selected codepage if the Font window is in Codepages mode.

We recommend setting the **Select encoding automatically** option as the default, because it covers most exporting situations very well.

Generation of Bitmaps

On the Macintosh Type 1 fonts must have associated bitmap fonts for the screen that are necessary for the system to store the font's metrics. TypeTool can automatically generate bitmap fonts with one of three rasterizers. Choose among the build-in rasterizer or ATSUI on Mac OS X. If you are using classic system and have ATM installed, then choose among the build-in and ATM's rasterizers:

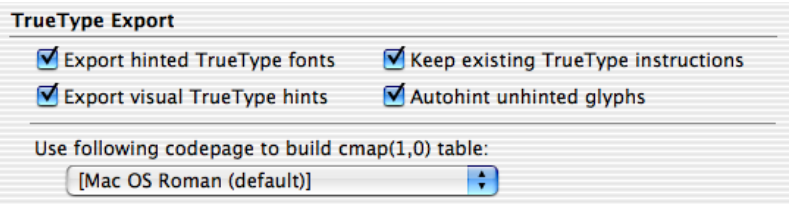
You can request several sizes of bitmap fonts in the Type 1 exporting options section:



Enter the point sizes of the bitmaps you want to export with the Mac Type 1 font, separating values by a space or comma.

TrueType Export Options

TrueType export options are concentrated in the **TrueType** page of the **Preferences** dialog box:


TrueType Export
Export hinted TrueType fonts
Keep existing TrueType instructions
Export visual TrueType hints
Autohint unhinted glyphs
Use following codepage to build cmap(1,0) table:
[Mac OS Roman (default)]

There are two groups of TrueType export options: **glyph export** (including control over instructions) and **Unicode export**.

Glyph Options

Glyph export options control how TypeTool exports glyphs and are related to the export of TrueType instructions:

Export hinted TrueType fonts	TypeTool will export TrueType instructions of any type (original or automatically generated) only if this option is on. To create a completely unhinted TrueType font switch off this option.
Keep existing TrueType instructions	If this option is on and the original TrueType instructions were saved when the font was imported, then TypeTool will try to restore the original instructions where possible. If you want to drop all the original TrueType instructions, switch this option off.
Export visual TrueType hints	TypeTool will convert visual TrueType instructions only if this option is on.
Autohint unhinted glyphs	If this option is on, TypeTool will try to automatically generate TrueType instructions for all unhinted glyphs.

How TypeTool Autohints TrueType Fonts on Export

When autohinting is allowed and TypeTool finds a glyph that has no TrueType hints of any kind (original hints imported from the TrueType font file or manually edited visual hints) it begins to make TrueType hints automatically. If this glyph has Type 1 hinting information, then TypeTool converts this information to visual TrueType instructions and converts the instructions to the TrueType hinting code. If Type 1 hints are not present then TypeTool automatically generates Type 1 hints as the first step, then converts the Type 1 hints into TrueType visual instructions and converts the visual instructions into TrueType native instructions.

Unicode Options

Most TrueType fonts use Unicode indexes for the glyph encoding information. Therefore TypeTool will always use the Unicode indexes that are set for the font's glyphs as the source of Unicode encoding information.

Use the following codepage to build a cmap (1.x) table

Encoding information is stored in TrueType fonts in a "cmap" table. Every Windows TrueType font contains at least two of these tables. One is the Unicode table and it "assigns" Unicode information to glyphs. The other is a single-byte table that is used by older versions of the Mac OS and by some non-Unicode-compatible Windows programs.

Use the control to select the codepage that will be used to build this table. There are two special options: Mac OS Roman (which is the codepage used on the Mac and the default choice) or Current codepage in the Font Window, which means that TypeTool will use the encoding (yellow zone) currently selected in the Font window.

Recommended Export Options

We recommend selecting different export options in different situations. In addition to the export options we also suggest particular import options that will produce the best results in specific situations:

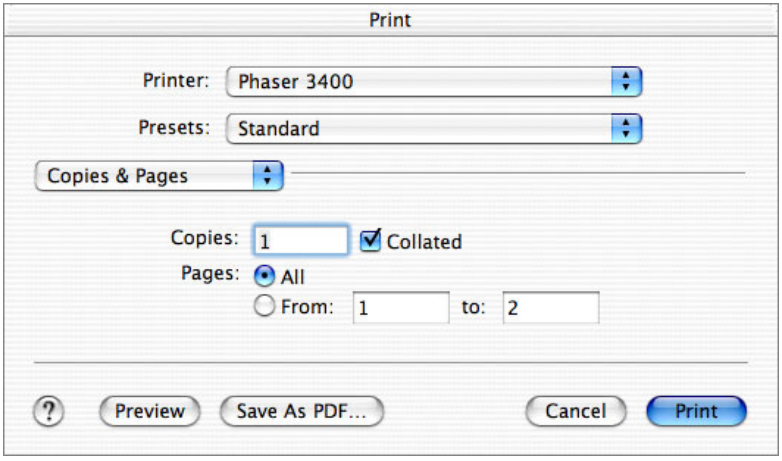
Source	Destination	Import options	Export options
TrueType	TrueType	Do not convert outlines, do not decompose, save hinting, do not autohint	All hinting options — on
TrueType	Type 1	Convert outlines, scale to 1000 UPM, do autohint, do not decompose, save hinting	Write PFM, AFM and INF files, Select encoding automatically. Before export, switch the Font window to the Names mode and select the desired encoding vector
Type 1	Type 1	Do not decompose, Generate Unicode	Write PFM, AFM and INF files, Select encoding automatically
Type 1	TrueType	Do not decompose, Generate Unicode	All hinting options — on

Of course you can choose other options, but when you just want to convert a font from one format to another these recommended combinations of import and export options will usually give you fonts that will work fine in most environments.

Printing Fonts

To print a font chart containing samples of all font glyphs select the **Print** command in the **File** menu.

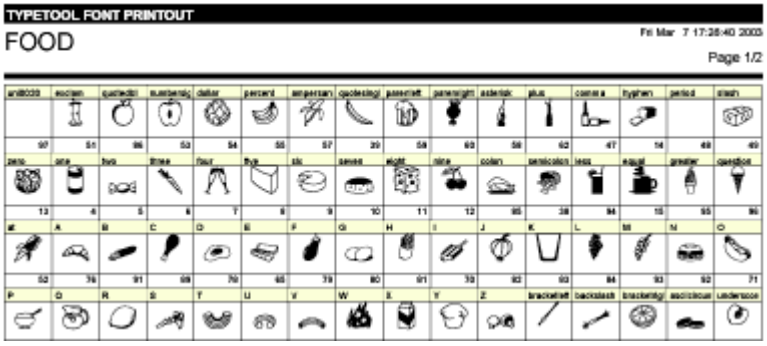
You will see the standard **Print** dialog box that will ask you to choose a printer and modify the printer settings:



In this dialog you can choose the range of pages you want to print.

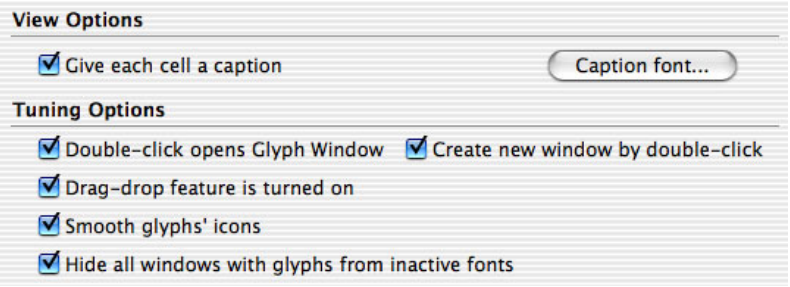
When you press the **Print** button, TypeTool will print a font table containing samples for all font glyphs, their names, and codes according to the current encoding and glyph cell caption settings.

Printout of the font chart:



Font Window Options

To change Font window options, select the Font Window page in the **Application>Preferences** dialog box:


View Options
Give each cell a caption
Caption font...
Tuning Options
Double-click opens Glyph Window
Create new window by double-click
Drag-drop feature is turned on
Smooth glyphs' icons
Hide all windows with glyphs from inactive fonts

Controls on this page are separated into two groups: View options and Tuning options.

View Options

Give each cell a caption

Use the check box to switch on or off the character cell caption. Use the **Caption font** button to select the font that appears in the caption of the cells.

Tuning Options

Double-click opens Glyph Window

Switch this option off if you don't want TypeTool to open a glyph in the Glyph Window when you double-click on the glyph cell.

Create new window by double-click

If this option is on, TypeTool will create a new Glyph window when you double-click a cell in the Font window characters chart. When it is off double-clicking a cell will create a new glyph window if there is none, but if there already is a glyph window then the glyph in it will be replaced by the glyph from the cell that was double-clicked.

Drag-drop feature is turned on

Switch this option off to switch off drag-drop editing in the Font window.

Smooth glyphs' icons

Activate this option to have glyph sample images in the Font window appear smooth. This increases visible resolution but requires more computing power. If you feel that the Font window is slow, try deactivating this feature.

Hide all windows with glyphs from inactive fonts

When this option is on TypeTool hides all Glyph and Metrics windows with glyphs from any font whose Font window is inactive. This feature helps to avoid screen crowding with many unnecessary windows.

The Font Header

Perhaps the most important information you need to define for a font is its header or font info data. This information is mainly used to properly register the font in the operating system and in any program that uses it.

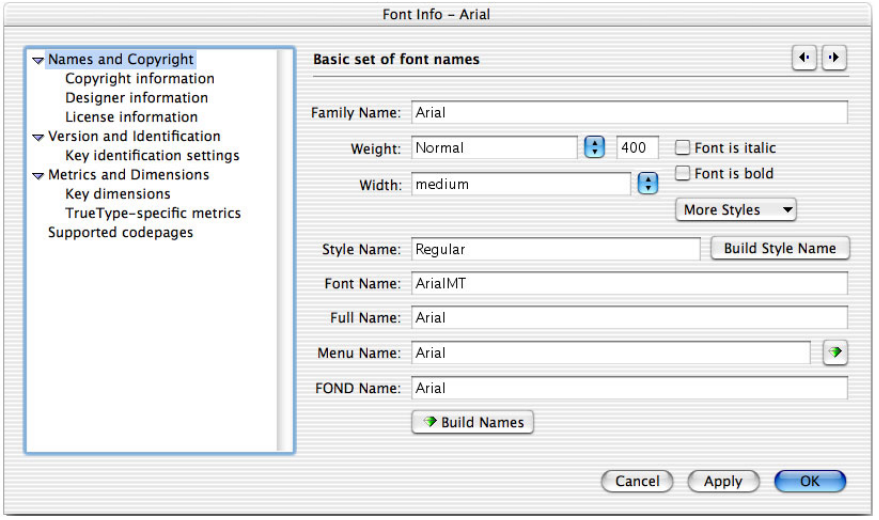
It is very important to carefully define all font parameters. Even the best-designed font is useless if it cannot be installed.

3

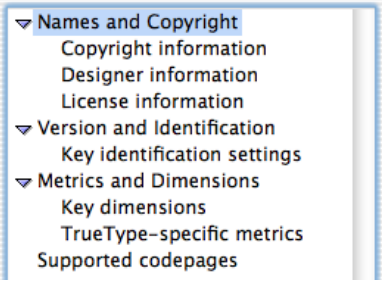
Font Info Dialog Box

The control center where you define font parameters is called the Font Info Dialog box and is accessible from the **File** menu or with the button on any Font window:

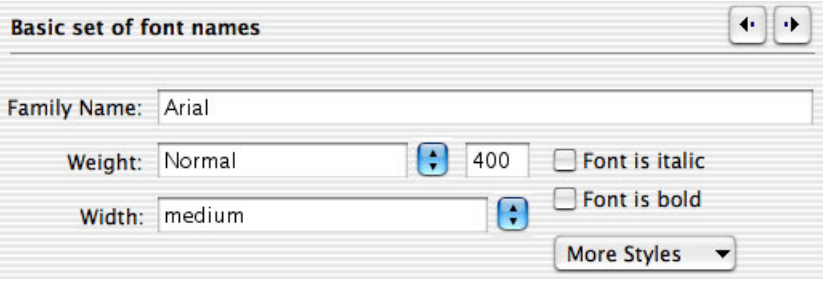
The Font Info dialog box consists of three parts:



At the left there is a page selection control where you can choose one of the sections in order to edit part of the Font Info information:



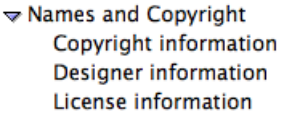
When you select one of the pages, it immediately appears to the right of the list:



Use the arrow buttons in the top-right area of the page to browse all available pages:

Alternately you may use the **Ctrl+Tab** and **Ctrl+Shift+Tab** key combinations to browse pages.

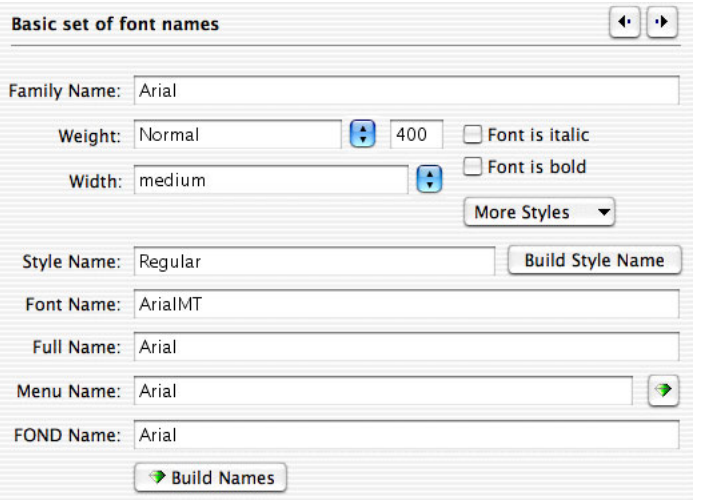
Font Names



The names section includes the most important font-registration information.

All programs use the information on this page to refer to a font. Be sure to enter all the values very carefully and use the automatic features where available.

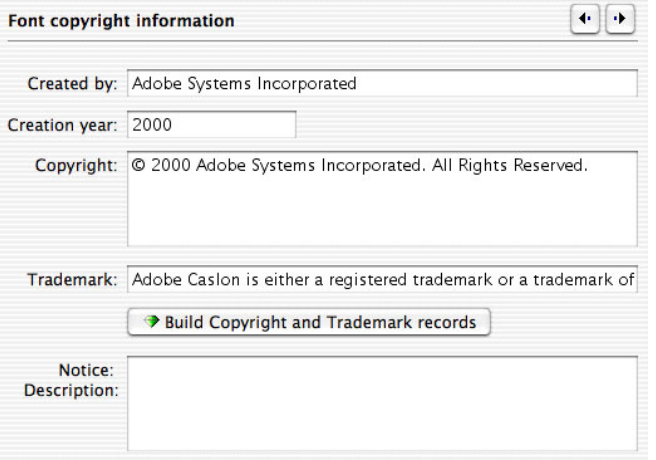
Basic Identification and Names



Family Name	The name of the typeface to which the font belongs. All fonts that are from the same typeface must have the same **Family Name** field. The Family Name is used as the root of the Font Name so we recommend that you fill in this field first.
Weight	Weight of the font. You may enter a custom value in this field or select one of the predefined weight names in the popup. Values in this list are sorted by increased weight value. Choose **Normal** or leave this field empty if you do not care about the font's weight.
Weight Value	Numeric weight value of the font. This number defines the font weight and is used by the operating systems to organize fonts to font families. TypeTool will fill it automatically when you select some Weight in the popup menu, but if you want you can customize it.
Width	The average width of the font's characters. Enter a custom value or select one of the predefined width values from the popup menu. Leave this field empty or select **Normal** width if you do not care about the font's width.
Font is Italic	Switch on this check box if you are creating an italic font.

Font is bold	The Font is defined as bold. Usually this checkbox is related to the Weight setting, but it is not required. For example, if you are making a family containing Light and Normal styles, you may need to mark the Normal style as Bold so you will not need to split these styles into two separate families.
More styles	Press this button to open a popup menu where you can select one of the additional font styles. This information is used only by TrueType fonts, but we recommend you always set it properly to simplify future font identification.
Style Name	Contains complete style information about the font. We recommend that you fill in the Weight, Width and Italic data, to automatically generate this field using the **Build Style Name** button and edit this field if necessary.
Build Style Name	Press this button to automatically generate the **Style Name** field. Style names are based on the Width, Weight and Italic information.
Font Name	PostScript name. This name will be used by a PostScript print driver to reference the font. Do not include spaces in this name.
Full Name	More detailed font name. It may include spaces as well as any other characters – this is the name that is exposed to users when the font is installed in Windows and sometimes in Mac OS.
Menu Name	The name used to access the font in applications. This name must not include style information (bold, italic or similar). The length of this field is limited to 31 characters for TrueType or single-master Type 1 fonts and to 7 characters for Multiple Master fonts. To ensure that the current Menu name is made properly, press the **Check** button.
FOND Name	This name is used by the Mac OS to organize fonts into font families. Windows does not use it. We recommend you fill in this name if you plan to use your font on Mac.
Build Names	Press this button to automatically generate the **Font Name, FOND Name** and **Full Name** fields. If you are creating a new font we recommend that you fill in the **Family Name** field, generate or manually fill in the **Style Name** field and press this button to create the Font, FOND and Full names. If necessary you can edit the names later.

Copyright Information



On the copyright page you can enter information about the creators of the font. If you have created a new font you should enter your copyright notice here. If you have edited an existing font that was not your creation you must not remove the information contained on this page, or you may violate copyright laws.

Created by	Name of the company or person that created the font. If you are creating a new font enter your name or the name of your company here.
Creation year	Year when the font was created. This is used by TypeTool to automatically fill in the **Copyright** field and is exported in TrueType fonts as the Creation year entry.
Copyright	Copyright message. Must include the © sign or the word "Copyright", the name of the company or person that owns the copyright and the copyright year. In Type 1 fonts this information is stored in the Notice entry and in TrueType fonts in the Copyright entry.
Trademark	Font trademark – used to save font's trademark notice.
Build Copyright and Trademark Records	Press this button to create the standard Copyright record based on the **Created By** and **Creation Year** fields.
Notice Description	Additional information that you want to include in Font Info. Exported in Type 1 fonts as the Copyright entry and in TrueType fonts as the Description entry.

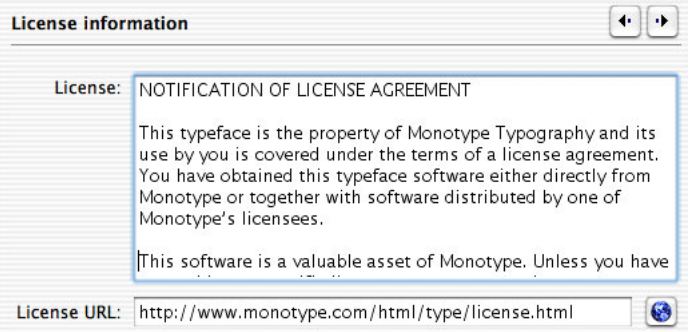
Designer Information

This page stores information about the font's designer. Do not modify this data if you open an existing font to modify for personal use.

Designer	Name of font designer
Designer URL	A new entry implemented only in TrueType format. It is the WWW link to the designer of the font.
Vendor URL	This TrueType-only entry shows the WWW link to the site of the font vendor.

Use the buttons to the right of the Designer URL and Vendor URL controls to open pages in your default Web browser window. This requires an Internet connection.

License Information


License information
License:
NOTIFICATION OF LICENSE AGREEMENT
This typeface is the property of Monotype Typography and its use by you is covered under the terms of a license agreement. You have obtained this typeface software either directly from Monotype or together with software distributed by one of Monotype's licensees.
This software is a valuable asset of Monotype. Unless you have
License URL:
http://www.monotype.com/html/type/license.html

License and License URL records are relatively new and have appeared only in OpenType specification version 1.3.

License	License description – contains information about how the font can be used
License URL	URL where additional license information can be found.

Use the button to the right of the License URL control to open the page in a Web browser window. This requires an Internet connection.

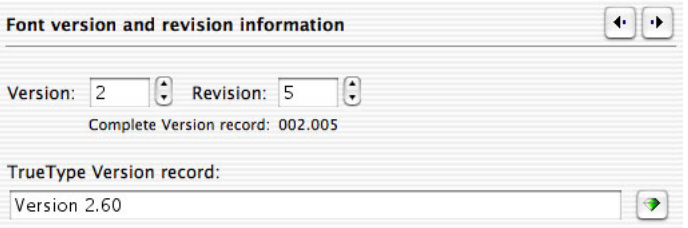
Font Identification

Version and Identification
Key identification settings

Sometimes the operating system or a DTP application needs to know what the font looks like. It may be necessary, for example, to properly substitute for a missing font with the closest lookalike.

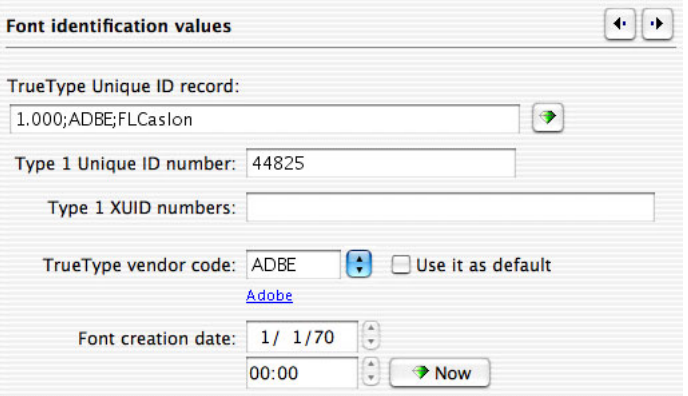
TypeTool supports some of the font-identification settings that are used in Type 1 or TrueType fonts.

Version Information



Version	Version of the font.
Revision	Revision of the font. Version and revision numbers are combined and build a complete version record that appears in Type 1 font headers.
TrueType Version Record	[name: 5] TrueType font version records have a different format. You may enter the TrueType version record here or just press the **Recalc** button at the right of the field to fill this record automatically. You must have the **Names** and **Copyright** pages filled in to use the automatic features on this page. Press the **Apply** button at the bottom of the dialog box to enter the new Font Info values into the font's header.

Basic Font Identification



TrueType Unique ID Record

This field is necessary to identify TrueType fonts. Usually it includes the creator's name, font family name and creation year. The format of this field is freeform, but we recommend that you use the button to fill this field automatically.

Type 1 Unique ID Record

An integer number identifying the font. Unique ID numbers must be registered with Adobe Systems. However, you may leave 0 in this field or enter a value from the users Unique ID zone (4000000 to 4999999). If you enter this value and plan to export Type 1 fonts, be sure not to have more than one font with the same Unique ID value because that may cause a problem with PostScript printers or Adobe Type Manager software.

Type 1 XUID Numbers

More advanced identification codes for Type 1 fonts. This number is used only in PostScript Level 2 printers. Please, refer to Adobe documentation for more information concerning the **XUID** field.

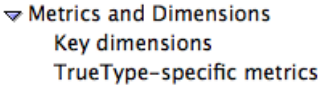
TrueType Vendor Code

An up-to-four letter length code that is assigned to most TrueType producers to identify their fonts. An *uppercase* vendor code must be registered with Microsoft or Apple. All registered Vendor codes known at the time of TypeTool's release are placed in the popup menu. If you want to identify yourself without registering you may enter a *lowercase* four-letter vendor code.

Below the vendor selection list you can see the full name of the registered vendor. Click the name to open the vendor's page in a Web browser.

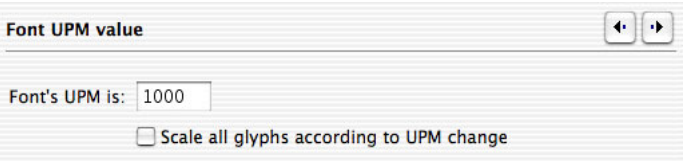
Use it as default	Check this option to use the current vendor code as the default in all new fonts. You may make your own code the default so you will not have to enter it every time.
Creation date and time	Use these controls to edit the font creation date and time. Press the **Now** button to set the current date and time.

Metrics and Dimensions



This page is used to set font dimensions that are used mostly to properly align text lines.

Font UPM Value



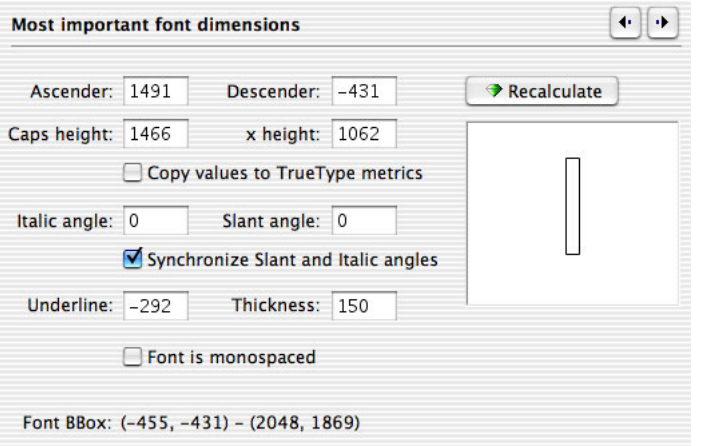
The most important field on this page is the **Font UPM**. Let's explain what UPM is and why it is so important.

The UPM (Units Per eM) is the basis of all font dimensions. The UPM is the number of font units that defines the font height and the coordinate grid on which the glyphs are drawn.

The bigger the UPM is the more coordinate space you have, so you can set more precise positions of points. For technical reasons in TypeTool the UPM is limited to 10000 units, but we strongly recommend you work with one of the standard UPMs. In Type 1 fonts the standard UPM is 1000 units and in TrueType fonts the UPM may be set to any value, but the recommended value is 2048 units.

If you change the UPM value in the Dimensions page of the Font Info dialog box this doesn't necessarily mean that the size of the characters will change. For example, if you change the UPM from 1000 to 2000 all the glyphs will now be half as big as they were before – because they are still dimensioned at 1000 UPM. You have to scale all the characters to fit them in the new UPM setting manually or you can switch on the **Scale all characters according to UPM change** check box and all the font data will be scaled automatically.

Basic Font Dimensions



The page has several editing fields with numbers and a sample window where an appropriate character is displayed to help set correct values.

Other fields on this page mean:

Ascender	Position of the font's ascender line. Usually this is the height of the lowercase 'b' character.
Descender	Position of the font's descender line. Usually this is the position of the bottom line of the 'p' character.
Caps height	Height of the font's uppercase characters. Usually the height of the 'H' character.
x height	Height of the lowercase characters. Usually the height of the 'x' character.
Italic angle	Actual italic or oblique angle for the font. The italic angle is measured in the counterclockwise direction, so the default value is -12°.
Slant angle	Type 1 fonts can be artificially slanted to get an "oblique" appearance while keeping the actual outlines upright. Enter a slant angle value (in degrees) here and check the result in the Preview panel.
Underline	This is the position of the middle of the underline line in your font.
Thickness	This is the thickness of the underline line.

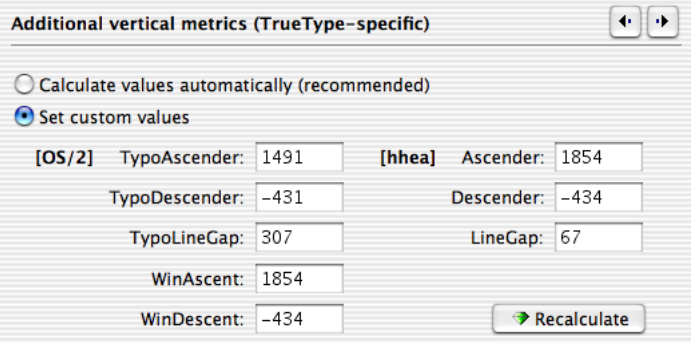
If you press the **Recalculate dimensions** button, TypeTool will automatically recalculate all the dimension values.

Ascender, Descender and Type 1 fonts

If you are making a Type 1 font you should set the Ascender and Descender values very carefully. In Type 1 fonts these values are used very directly to calculate interline spacing. It is usually necessary to set the Ascender value higher than actual height of the "ascender" 'b' character, to have some additional space between lines.

If you are designing a fixed-width font press the **Font is monospaced** check box.

Advanced Vertical Metrics



In TrueType font files vertical metrics can be stored in the OS/2 and hhea tables. Different programs and operating systems use vertical metrics from these tables. Windows usually uses data stored in the OS/2 table while the Mac OS uses only data located in the hhea table.

It is important to correctly define all vertical metrics if you want your font to align properly. In most cases TypeTool can calculate vertical metrics according to the system recommendations, but in some cases you may want to customize these values.

We recommend you generally leave these values untouched in an existing TrueType font. Of course, if you perform heavy modification of the font you will need to update the advanced vertical metrics.

If you want TypeTool to automatically calculate all vertical metrics, select the option **Calculate values automatically**.

If you want to customize values, select the **Set Custom values** option and edit data in the fields below. Note that if you choose Set Custom values but leave all data unchanged, TypeTool will restore the original vertical metrics data from the imported font and the new updated font will align exactly as the original one.

Here is a description of each value:

Typo Ascender	This is the typographically correct ascender value. It is the topmost line of lowercase characters, usually, the topmost line of the 'b' character.
Typo Descender	The same as Typo Ascender, but for the lowest line. Usually it is equal to the bottom line of the character 'p'.
Typo Line Gap	"Typographically" correct line gap value (distance between bottom line of the upper line of text and top line of the lower line of text).
WinAscent	[OS/2] This value defines the topmost line of all important characters in the font. "Important" characters are all non-exceptional characters. For example, if most of the characters have the topmost position at 900 font units and one, not often used character, has it at 1300 font units, it's a good idea to set WinAscent at 900 units. Note that in most cases portions of the characters that are above the WinAscent value will not appear on the screen or print on some printers. Please note that WinAscent is NOT a typography ascender, usually measured as the topmost line of lowercase characters. It is mostly a technical parameter used by the rasterizer to allocate vertical space to render characters.
WinDescent	[OS/2] The same as WinAscent, but for the lowest line of all "normal" characters.
Ascender	[hhea] This value is used by the Mac OS in about the same situation as Windows uses the WinAscent value from the OS/2 table – to define the topmost position of all important glyphs.
Descender	[hhea] In short: the Macintosh version of the Windows WinDescent parameter. If there are any pixels below this line the glyph will be squashed in the vertical direction to match metrics defined by the Ascender and Descender parameters.
Line Gap	[hhea] This value is used by the Mac OS to compensate Ascender and Descender values and calculate the correct distance between baselines of the text. Refer to the formulas below to see how baseline-to-baseline distance is calculated.

Baseline-to-baseline distance calculation

Windows:

Windows Metric	OpenType Metric
ascent	WinAscent
descent	WinDescent
internal leading	WinAscent + WinDescent – UPM
external leading	MAX(0, LineGap - ((WinAscent + WinDescent) - (Ascender - Descender)))

BTBD = ascent + descent + external leading

It should be clear that the "external leading" can never be less than zero. Pixels above the ascent or below the descent will be clipped from the character; this is true for all output devices.

Macintosh:

Macintosh Metric	OpenType Metric
ascender	Ascender
descender	Descender
leading	LineGap

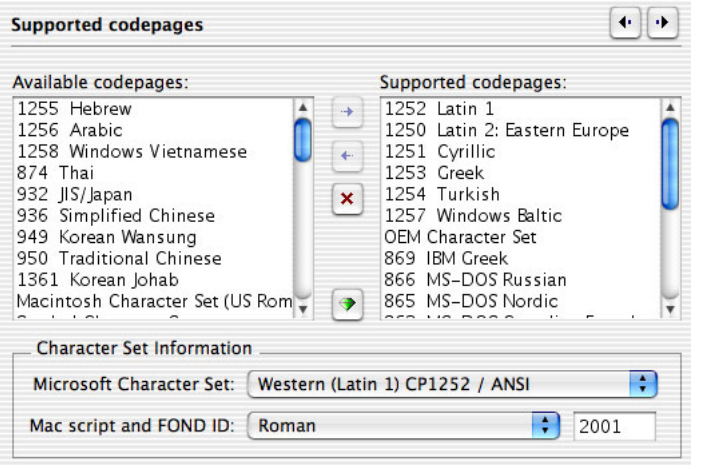
BTBD = ascender + descender + leading

Encoding and Unicode

As we mentioned earlier, fonts may have very many characters and support a lot of different languages. To tell the operating system what codepages the current font can support, you set the codepages information.

TrueType and Type 1 fonts use different methods to identify what codepages a font supports. In TrueType fonts you can identify all the supported codepages by setting bits in a special field of the font header. In Type 1 fonts you select only one codepage (actually, encoding vector) and it must be compatible with the actual font encoding.

Supported Codepages



The operating system needs to know which codepages a TrueType font can support. To set this information you select all the codepages that this font can "cover" from the list of standard codepages that are available to the operating system.

To select the supported codepages automatically press the **Auto** button. TypeTool will analyse the Unicode information available in the font and will automatically detect which codepages this font can support.

To add a codepage to the list of supported codepages select a codepage in the left list and press the **Add** button.

To remove a codepage from the list of supported codepages select a codepage in the right list and press the **Del** button.

To reset the list of supported codepages, press the **Reset** button.

The Meaning of Supported Codepages

In Mac OS prior to 8.5 and Windows 3.1x this information is not used.

In Mac OS starting from 8.5, Windows 95 and Windows NT a font that has more than one standard (Mac OS Roman or 1252 Latin 1) codepage supported will appear as a font available for different scripts. So, if, for example, you set Latin 1 and Cyrillic codepages for a font with the name MyFont, in Windows 95 (and NT) it will appear as MyFont (Western) and MyFont (Cyrillic).

Type 1 Character Set

Type 1 fonts do not have such extensive support for multiple codepages. The character names they use to identify characters are mapped to codes through the encoding vector. There is one parameter that is used to tell Adobe Type Manager (used to support Type 1 fonts in Windows) how to interpret the encoding vector. This is the Microsoft Character set.

Some of the values for Microsoft Character set:

ANSI	The Font has all the characters necessary to represent the standard Windows Latin 1 character set. No reencoding is necessary.
Symbol	The font is symbolic with a custom encoding vector. It should appear as Symbol in Windows applications and the font's own encoding vector should be used to access characters.
ShiftJIS	This is a Japanese font that includes Kanji characters.
OEM	The font has MS DOS characters. This setting is rarely used in Type 1 fonts.
Bitstream	This is a normal text font, but it has its own encoding that should be used to access characters. This setting is highly recommended for all text fonts with a non-standard encoding vector.
Arabic	The font has Arabic encoding.

Other values cover more codepages that may be supported by the font. Choose the codepage that is the default for your font.

The Macintosh Script and FOND ID

A Macintosh script is used to calculate the correct value of the FOND ID parameter used in Macintosh font suitcases to differentiate fonts. The Script is necessary to properly identify a font in multiple-language non-Unicode programs.

You can use the list of all the possible Macintosh Scripts to automatically calculate the FOND ID in the correct range (TypeTool will generate a random value) or you can manually enter the FOND ID and TypeTool will automatically select the script related to that value.

You may change this field later when exporting your font in any of the Macintosh formats. And certainly this field is not needed when saving the font for Windows.

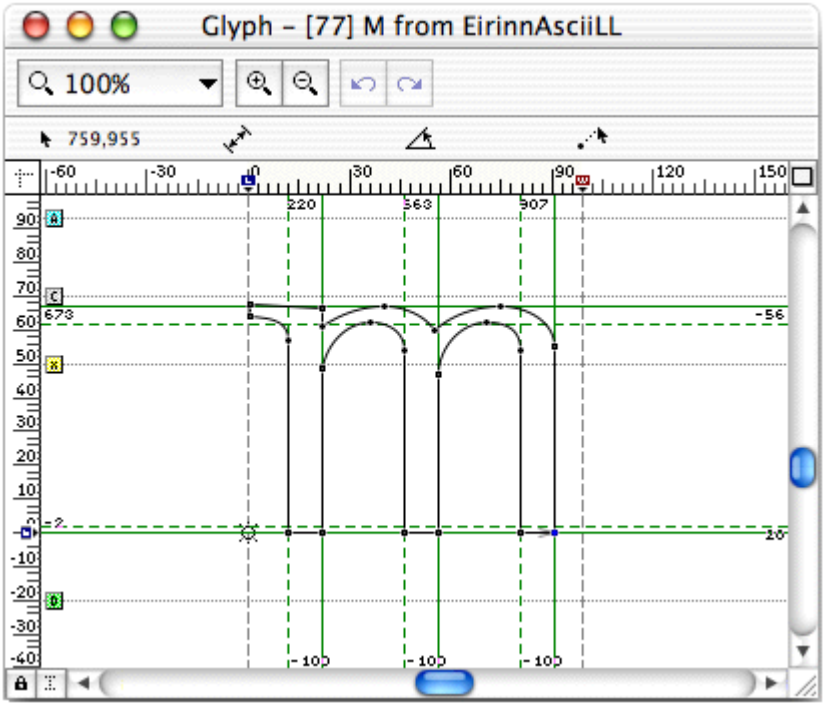
The Glyph Window

The Glyph Window is a standard tool in all FontLab based applications. It is a universal and very powerful contour-editing module that also allows you to perform many font-specific operations.

4

Glyph Window Contents

Open the Glyph Window by double-clicking any character sample in the Font Window or Metrics Window.



The Glyph Window

The Glyph Window has the following parts:

- Toolbar area
- Toolbar area expand/collapse button
- Editing Field
- Top and left rulers
- Left-Top box
- Scroll Bars
- Lock button
- Meter panel button

The *Local Toolbar* is the command center of the Glyph Window. You can switch it on and off by pressing the small button at the top of the vertical scroll bar: . There is a popup menu with zoom selection and two pairs of buttons:

The first pair of buttons is used to select the zoom mode.

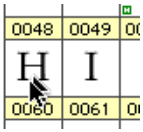
The second pair of buttons is used to apply **Undo** and **Redo** operations.

Below the Local toolbar and at the left of the window you may see rulers that are used to preview positions of various structures in the glyph space. You may switch the rulers on and off with the **View>Rulers** command or using the context menu that appears if you **Ctrl**-click one of the rulers.

Selecting a Glyph for Editing

You can open a glyph in the Glyph Window using any of the following methods:

- Double click the glyph's cell in the Font Window to open it:



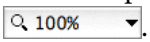
  If you already have an open Glyph Window with a glyph from the same font, the new glyph will be opened in the same Glyph Window (replacing the previous glyph). Hold the **CMD** key down when you double-click on the glyph cell to open it in a separate Glyph Window. Note that if this method doesn't work it usually means that it is switched off in the Font Window page in the Preferences dialog box.

  You can force TypeTool to **always open a glyph for editing in a new Glyph Window**. Use the **Create new window by double-click** option in the Font Window page to activate this feature.

- **CTRL**-click in the Font Window and select the **Edit in New Window** command to open the selected glyph in a separate Glyph Window.
- Click on this button in the bottom left area of the Glyph window to unlock the keyboard and click on the keyboard to open the character for editing. You can also type the glyph name if you know it. Use the **CMD-L** shortcut to quickly lock and unlock the keyboard.
- Click the mouse button on the glyph selected in the Font Window and drag it into any Glyph Window.
- Press the or buttons in the Tools toolbar to move to the previous or next glyph. The Keys ',' and '.' are equivalents for these buttons.
- Select the **Choose Glyph** command in the **Glyph** menu and find the glyph that you want to open.

Changing the View in the Glyph Window

Use the zoom mode and scroll bars to change the view in the editing field of the Glyph Window. By using the scroll bars you can scroll the viewing field of a symbol. With the zoom mode you can define how the symbol coordinates are converted to screen coordinates and vice versa. If you choose a bigger zoom mode you will see a more detailed symbol and you can do the editing operations more precisely. However, in the larger zoom modes not all of the symbol will lie inside the editing field so you will have to use the scroll bars to see the different parts of the symbol.

There are fixed zoom modes and custom zoom modes. You can select one of the fixed zoom modes in the **Zoom** popup menu located in the upper part of the Glyph Window:  . When you choose a fixed zoom mode TypeTool will return to this glyph mode on every **Zoom Out** command.

To magnify part of the glyph, select the **Zoom** tool (button on the toolbar or the + key on the keyboard) and declare a custom zoom mode. This mode is temporary and you can always return to the previously selected fixed zoom mode by clicking on the button (or by clicking on the - key).

Alternative keyboard shortcuts are:

Command+Space, then click	Zoom in
Command+Option+Space, then click	Zoom out
Command+1,2,3,4,5,6	Set fixed zoom mode from 6.25% to 200%
Space and drag	Scroll (hand cursor appears)

After you select the zoom tool , move the mouse pointer to one of the corners of the rectangular area that you want to zoom on and press the mouse button. Then, holding the mouse button down, define the zoom-in area by dragging the cursor to form a rectangle. Release the button and the new zoom mode will be selected.

Quick Zoom Selection

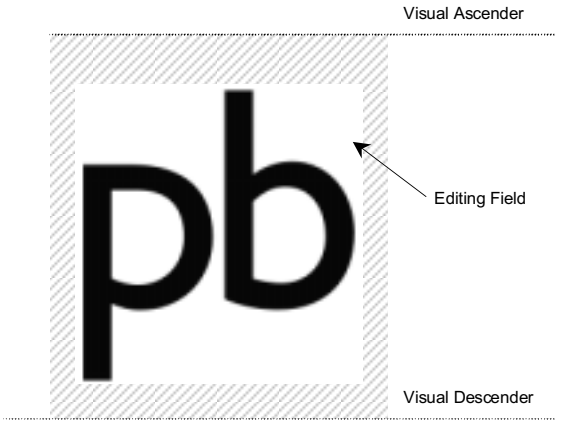
You can quickly change the zoom mode of the Glyph Window by selecting the **Zoom In** or **Zoom Out** command from the **View** menu. Alternatively you can click the **Z** key for zooming in or the **X** key for zooming out.

This command increases or decreases the zoom mode by a factor of two. If the mouse cursor is in the editing area of the Glyph Window the new zoom mode will be centered around the cursor position.

These keys are active even when you drag something with one of the editing tools.

Vertical Alignment Options

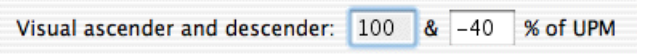
When you select 100% as the zoom value, TypeTool needs to choose a scaling factor to fit the font unit space in the Glyph Window. Two vertical levels in the font space define this scaling: Visual Ascender and Visual Descender:



When you select 100% zoom, it means that Visual Ascender is fitted to the top of the editing field and Visual Descender to the bottom.

The same values are used to build the icons that you see in the Font Window.

To set Visual metrics, use the **Glyph Window** page of the **Preferences dialog box** (**Application>Preferences** menu):

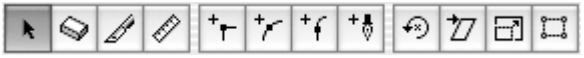


Values are measured in percentage of font UPM, so –20% is –200 if UPM is 1000 and –410 if UPM is 2048.

Bezier Drawing Tools

All operations performed with the editing tools can be undone with the **Undo** command of the **Edit** menu, or just by clicking the **Undo** button on the toolbar at the top of the Glyph Window. You can undo up to 200 operations. All undone operations can be redone with the **Redo** command of the **Edit** menu or with the **Redo** button on the toolbar.

Use twelve different Edit Tools. You can easily choose one of the tools using the Tools toolbar:



Alternatively you may use the keys from 1 to 8 to quickly select edit tools:

	1 Edit	Main tool, used to drag objects on the editing layers and perform other operations.
	2 Eraser	This tool is used to quickly remove unnecessary nodes.
	3 Knife	Tool to insert nodes and break outlines.
	4 Meter	A special tool to measure distances between nodes, etc.
	5, Add Corner, **6, Add Curve,** **7 Add Tangent**	Tools to create new contours or insert nodes.
	8 Bezier Drawing	Tool to draw the contour with the Bezier curves.
	Rotate, **Slant,** **Scale,** **Transform**	Tools to quickly transform outlines.

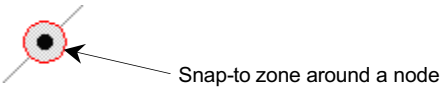
Temporary Activating the Edit Tool

You can temporary activate the Edit tool from any other tool. Just click on the **Cmd** key on the keyboard. Second click on the **Cmd** key will return the tool you were using before.

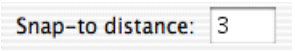
Snap-to Distance

In the following sections we will discuss how to use the Edit tools to modify the outline and other editing layers. All other tools will be explicitly named.

When you need to select a node or any other object on any of the layers, you need to click it with the mouse. You don't need to click the object precisely, but you must be within a certain distance, which is called the "snap-to distance".



Snap-to is used when you select an object for which the feature is allowed. By default the snap-to distance is set to 3 screen pixels, but you can change it on the **Glyph Window** page of the Preferences dialog box:



Editing Layers

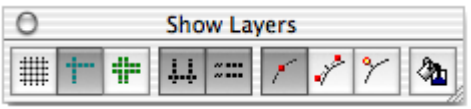
In TypeTool every glyph contains several editing layers. Some of them are used when the font is exported, others are TypeTool-only and are used to help you work with the glyph. Below is the list of all the layers that you can see in the Glyph Window. Later we will describe them in full detail.

	Outline	Main layer containing the glyph's outline. There is no button to switch the outline layer off.
	Grid	Regular grid which helps to align the outline
	Guidelines	Horizontal, vertical and/or diagonal guidelines
	Hints	Type 1 hints – pairs of two vertical or horizontal lines set at a fixed distance
	Glyph metrics	Glyph metrics – left and right sidebearings and a baseline
	Vertical metrics	Font's vertical metrics, such as ascender, descender or cap height

You can control the layers' appearance and features with the **View** menu:

Show layers	Lists all layers (and a few other options) and lets you switch them on or off. You cannot switch off (hide) the outline layer.
Snap to layers	Controls which layers have the "snap-to" feature activated.

If you frequently need to switch some layers on and off you can use the **View>Toolbars** menu to open the **Show Layers** toolbar:



In the following sections we will describe all the editing layers and their modification and control.

Outline Layer

The Outline layer is the most important of all the layers. It stores information about the glyph shape while all the other TypeTool editing layers and most of the tools are designed to help you create good outlines. Before we turn to the outline editing tools let's talk about outline structure.

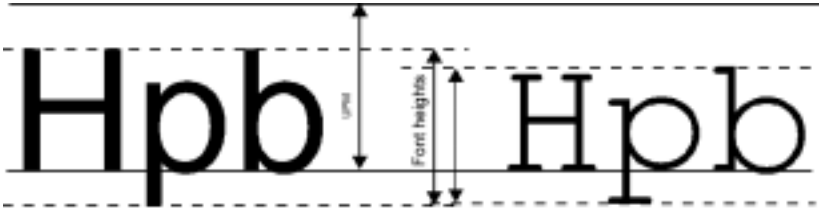
Units of Measurement

The coordinates of any object in the font are presented in a standard measurement system. One unit of this system is called a font unit. The scale of font units used in a particular font is the *Units Per eM (UPM)* font parameter. The program that scales the font knows the UPM value of the font and can use it to properly scale it. To get a text string of the same visible size a font that has a larger UPM value must be scaled with a smaller scale factor.

Usually Type 1 fonts have a UPM value of 1000 font units. Therefore, to get a text string with a height of 100 pixels (assuming that we use a raster output device) we would scale this font with a scale factor of 10%.

TrueType fonts usually have a UPM of 2048 units. So to get a text string of 100 pixels we would scale this font with a scale factor of 4.9%. Essentially the UPM is nothing more than a base value used for proper font scaling and as a base value for the font unit.

A more “graphical” font parameter is the *font height*. The font height (measured in font units) is the measurement of the font that is used to align strings in text. It is important not to confuse UPM and font height. UPM is just a scaling base, and, for example, all Type 1 fonts have the same UPM of 1000. Font height depends on the font design and may be different:



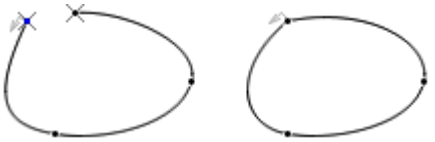
The font height can be defined as the distance from the bottom of a letter that is partially located below the baseline, like the ‘p’ character, and the topmost point of an uppercase character, like ‘H’, or a tall lowercase character, like ‘b’. Sometimes a font contains special glyphs that can be taller than ‘b’, like an integral sign, but usually these glyphs are not counted when font height is measured.

Contours

The most important and most complex information in a font is the glyph's shape. All glyphs are defined as a series of contours. All contours consist of a series of graphical primitives: straight lines and curves. Nodes – points that know their type – define all graphical primitives.

Open and Closed Contours

Contours may be open or closed:

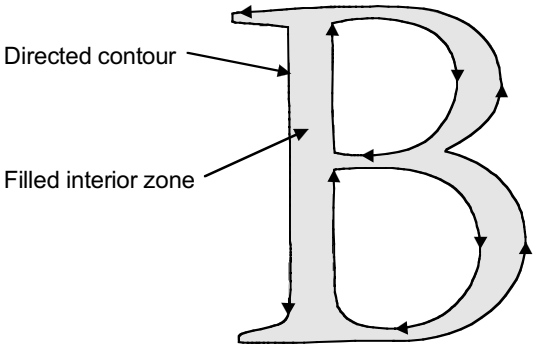


All known font formats require contours to be closed, but during outline editing it may be useful to have some contours in an open form and later connect them to each other to build final closed contours.

In TypeTool it is very easy to open closed contours or to close open contours. It is also possible to customize the appearance of the open contours: they may be automatically filled (they are automatically closed by an invisible straight line that connects their starting and ending points) or not and their starting and finishing point may be optionally highlighted.

Filled and Unfilled Contours

Contours can be of two types: black or white. They can also be of two directions: clockwise or counterclockwise. The basic rule that applies to Type 1 fonts is simple: clockwise-directed contours are white and counterclockwise contours are black. A simpler form of the rule, known as the rule of the *left hand*, is: if you face along the direction of a contour, black (fill) will be on your left side.



In the TrueType specification the opposite is the case, so a contour is filled on the right hand side. However, not all TrueType rasterizers require glyphs to follow this rule, so it is recommended, but not necessarily required, that you reverse contour directions when you are converting Type 1 fonts to TrueType.

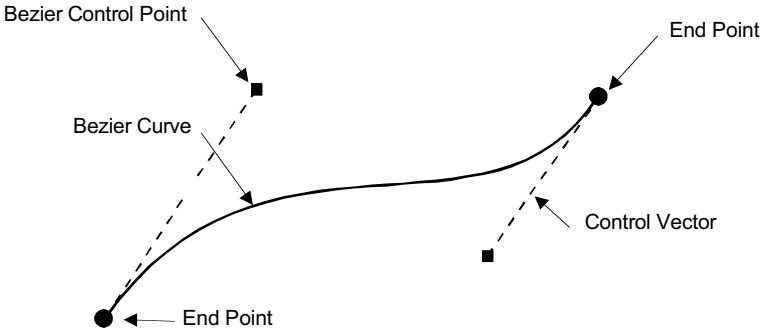
Startpoint and Closepath

All contours have a *startpoint*. The startpoint is the first node of the contour. The last node of the closed contour is automatically connected to the startpoint with a straight line, which is called *closepath*. The color of the start point in the Glyph Window is **blue**.

A contour-direction mark may appear on a startpoint:

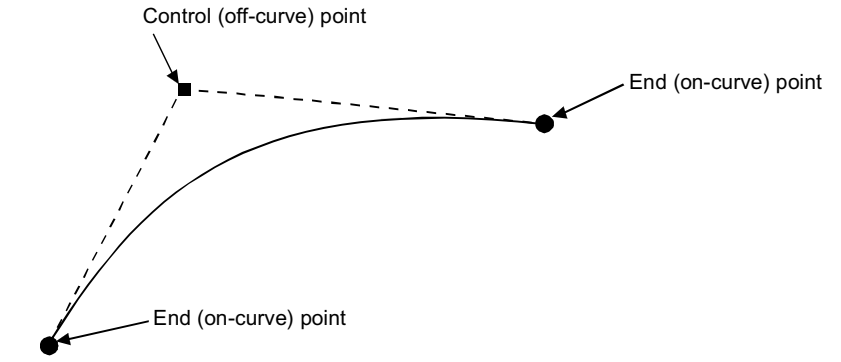
Curves and Lines

Graphical primitives are of three types: straight lines, Type 1 curves or TrueType curves. Straight lines (we'll call them *vectors*) are just straight lines that connect two sequential nodes. Type 1 curves are *Bezier curves* (3rd order, cubic b-splines). To modify the form of the curves two additional sub-nodes are used:

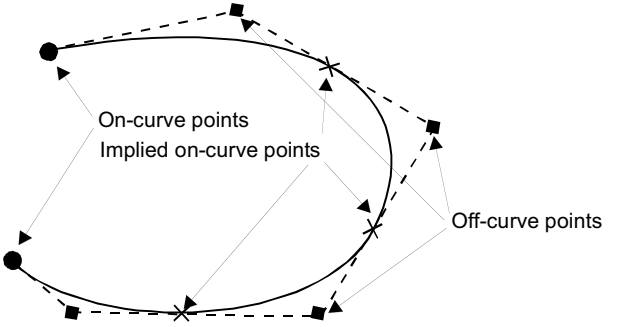


These *sub-nodes* are called *control points* and the vectors that connect the control points with the curve's ends are called *control vectors*. In the Glyph Window vectors (straight lines) end in **square** (in black-white mode) or **red** (in color mode) dots and curves end with **round** or **green** dots.

TrueType curves are 2nd-order curves (quadratic *b-splines*) that have one control point, called the "*off-curve*" point:



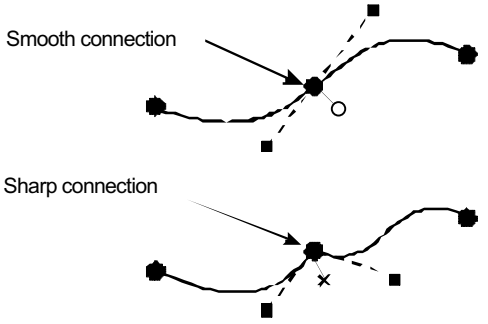
Some TrueType curves may appear linked together and form a long "*supercurve*" where intermediate on-curve points do not really exist, but are implied by the rasterizer:



TrueType curves end with points that look exactly like vector points. Off-curve points of the TrueType curves have a "**plus**" (in black-white mode) or **light-blue** (in color mode) appearance.

Connections

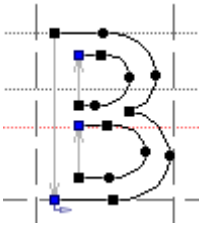
The type of connection between graphical primitives is very important if you want to keep the contour smooth at appropriate nodes. There are two types of connections: sharp and smooth. At a sharp connection, the two connected graphical primitives (curve and curve or vector and curve) are absolutely free in their angle relative to each other at the connecting node. At a smooth connection, the direction of the vector and the control vector of a curve or the control vectors of two sequential curves are kept collinear (lie on the same straight line). I.e. the angle between the two primitives at the node is fixed at 180 degrees.



It is very important to maintain the smoothness of the glyph's contours at the appropriate places. Small corners (sharp connections that are invisible when glyphs are small) become visible (and ugly) when you print large text. Furthermore, rasterizing programs that convert outline glyphs into bitmap images on paper do not like outlines where sharp connections are present in places where the outline should be smooth.

Outline Appearance

You can view an outline in contoured or filled mode. These modes are equivalent for all editing operations, but the filled mode is a little bit slower. However, in the filled mode you always see how the glyph will look in the resulting font. Switch between modes with the button on the **Show Layers** toolbar.



Outline mode

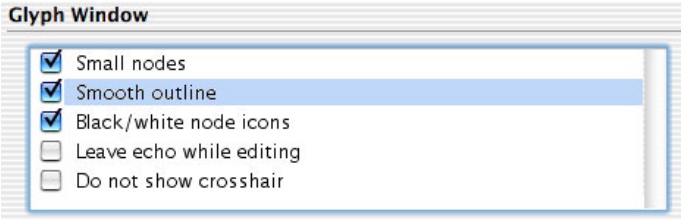
Preview mode

Smoothed Contour

By default contour is rendered with black color and sometimes it may result in jaggies:

Optionally you can smooth the contour appearance on screen, which will result in a much smoother outline appearance:

To smooth contours, use the Glyph Window page of the Preferences dialog box (**Application>Preferences** menu command):



You can use the **Apply** button in the bottom-right of the Preferences dialog box to check the result of the changes you make in Glyph Window options.

➲ **Note:** If your computer is slow and a contour is complex, smoothing the outlines may degrade the performance of the editing tools. Turn it off in this case.

High-quality Preview

No matter which mode is active you can quickly view a high-quality preview of the outline by pressing the ‘`’ key on the keyboard. Until you release the key you will see a high-quality preview of the outline. Note that you can use the ‘<’ and ‘>’ keys to browse characters without releasing the ‘`’ key.

Outline Preview Options

You can choose various options for previewing the contour layer. Use these buttons in the Show Layers toolbar:

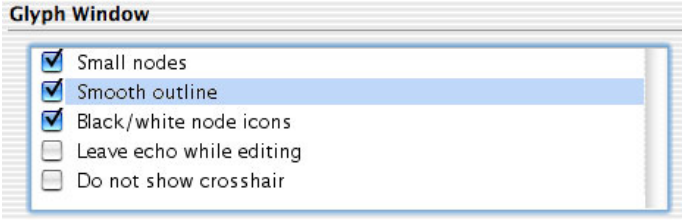
	Nodes	To show nodes or not.
	Control vectors	To show curve control vectors or not.
	Connection mode	To show connection mode marks.

All contour presentation modes may be combined in any way.

A few more notes about outline appearance:

Selected parts of an outline appear red in color. Selected nodes are marked as red rectangles and they are visible even if non-selected nodes are hidden.

Many options related to outline appearance can be customized on the Glyph Window page of the Preferences dialog box (**Application>Preferences** menu command).



Here is a list of the available options and a description of the features they control:

Small nodes

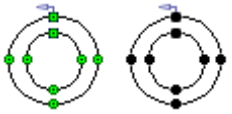
Nodes may be small or large:

Smooth outline

Allows one to select between standard and smoothed rendering of the outline:

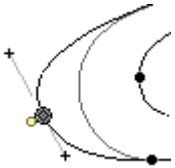
Black/white node icons

Nodes may be colored or black/white:



Leave echo while editing

Echo shows how the outline looked before you moved a node:



Moving Nodes

The most important editing operation is the modification of the contours that build each glyph. You can modify contours in three ways: moving nodes, editing graphical primitives using non-node editing, and selecting several nodes and moving them together.

To move individual nodes:

1. If nodes are hidden, make the node that you want to edit visible: switch nodes on with the **View>Show Layers>Nodes** menu or click near the node to make it visible. If you missed and an incorrect node is highlighted, use the **PAGE DOWN** and **PAGE UP** keys to move to the correct node:

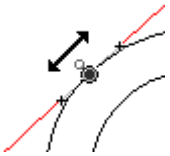
2. Drag the node to the new place. It will stick to other layers if they are visible and snap-to those layers that are activated (**View>Snap-to-layers** menu).

   Hold down the **SHIFT** key to constrain the direction of the node's movement in 15-degree increments and to snap the cursor to the original node's position.

Options

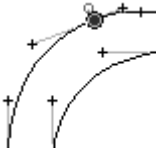
If you are moving a node that is connecting two Bezier curves you have the following options:

1. If the connection of the curves is smooth, press the **Shift** key before clicking the node to constrain movement to a line between the curves' control points:

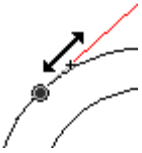


2. If the connection is sharp, press the **Option** key at any time while dragging the node to move it without the adjacent control points:

3. If the connection is smooth, press the **Option** key before moving the connecting node to keep the connection's curvature optimised. Hold down the **Cmd** key to involve all 4 control points in the process:



4. When you are editing control points of a Bezier curve press the **Shift** key before clicking the button to keep the direction of the control vector unchanged.



5. If you are moving a control point of a curve with a sharp connection, press the **Option** key to temporarily change the connection type to smooth, so that the adjacent control vector will be collinear.

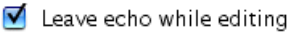
Do not forget that you can press the ‘`’ key at any time to get an instant high-quality preview of the glyph outline as it will print:

Normal outline

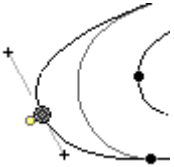
High-quality preview

Outline Echo

If you want to see how the outline looked before you moved a node, switch on the Echo mode. Open the **Glyph Window** page of the **Preferences** dialog box and switch on this option:



This is how editing field will look when echo mode is on:



Using the Keyboard

You can use the keyboard to move nodes and to select a node for editing:

Arrow keys	Every click moves current node by one font unit.
Shift+Arrow keys	Every click moves a node by 10 font units.
Cmd+Arrow keys	Every click moves a contour that contains a node by one font unit
Cmd+Shift+Arrow keys	Every click moves a contour by 10 font units.
Page Up	Selects the previous node for editing
Page Down	Selects the next node for editing
Tab	Alternates between the node and Bezier control vectors
Esc	Drops the selection of the current node

➲ **Note:** You can make a line or a curve a current object and arrow operations will move it as a whole. Just click on a curve or a line with the Edit tool and it will be highlighted by a pair of short lines:

Non-node editing

Sometimes you may want to modify a contour in a more flexible way than by moving nodes. For example, to adjust the shape of a curve in node editing you would usually make the control points of a curve visible and move them to modify the curve. A more intuitive way would be to "grab" the curve somewhere between the nodes and move this imaginary "inside" point. The curve's shape changes accordingly. We call this method "*non-node editing*". This means that you can move not just nodes, but every point of a glyph's contour. You can even switch off nodes and still be able to edit the contour as you wish.

To modify a curve or vector using the non-node editing method:

1. Move the mouse cursor onto the place on the curve (vector) that you want to move.
2. Press the mouse button. You will see a small color point that will show you the temporary point that you are moving.
3. Drag the mouse and observe how the shape of the curve changes. After a few experiments (which can be undone) you will have enough experience to use this method of editing.

Several notes that you should remember:

1. In non-node editing, guiding objects are not sticky. So, temporary points do not snap to the grid, guidelines, hints or anything else.
2. If you choose a temporary point near one of the ends of a curve, you will move that end, not just change the curve's shape. This is a useful method to locate a curve's endpoints.
3. When you press the mouse button to begin non-node editing you will see that the endpoints of the curve as well as the control vectors appear, simplifying the editing of this primitive.
4. If you want to highlight the line or curve but don't want to modify it, hold down the **CMD** key while clicking on the line or curve.

If you drag a "point" on a curve, its control vectors may change direction:

To fix the direction of the control vectors, hold down the **OPTION** key and double-click the node. You will see the connection mark turn yellow. **OPTION**+double-click it again to remove the fixed state. Alternatively you can **CTRL**-click the node and use the **Connection>Fixed** option to control this feature.

Changing Connection Type

The type of connection between graphical primitives is very important in maintaining the smoothness of contours. Connections can be of two types: smooth and sharp.

To make connection marks visible click on the button on the Show Layers toolbar.

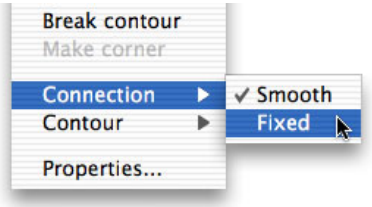
If a connection is smooth the direction of the adjacent curve control vectors or of the curve control vector and line is collinear and the contour is smooth at the connection.

To change the type of connection

1. Make the node visible.

2.1 Double-click the node

2.2 CTRL-click the node and select the connection type in the popup menu:



Deleting Nodes

To delete nodes using the Edit tool:

1. Begin moving the node by dragging it.
2. While dragging, **Ctrl**-click the mouse button. The node will be removed.

---

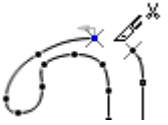
1. Move the mouse cursor onto the node and **Ctrl**-click the mouse button.
2. In the popup menu choose the **Delete node** command.

➲ **Note:** If you **Ctrl**-click the mouse button while editing the curve using the non-node editing method or while you are moving the control points of a curve the curve will not be removed. Instead it will change to a straight line.

You can use the **Edit>Delete** and **Edit>Cut** commands to remove nodes and other selected elements of a glyph as well. Usually, deleting a selection breaks the contours.

Deleting Lines and Curves

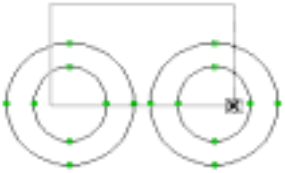
You can delete a whole line or curve with the **Knife** tool. Activate the Knife tool with the button on the Tools toolbar or click '3' on the keyboard. Press the **Option** key and click on the line you want to delete. Note that with this method you will always break the outline:



You can use the **Edit>Delete** (the **Del** or **Backspace** key) and **Edit>Cut** commands to remove the active curve or vector as well.

Eraser Tool

The **Eraser** tool can be used to quickly remove nodes. Sometimes this is necessary, for example, with contours from an auto-tracing program. The eraser tool can work in two modes: like a standard eraser or as a rectangle eraser. In the first mode, all nodes that are inside the rectangle of the eraser mouse cursor are deleted. In the second mode, you define a rectangle by clicking and dragging (as when you select nodes with the **Edit** tool or change the zoom of a Glyph Window) and all the nodes inside the rectangle are removed.



The first (eraser-like) mode is the default for the **Eraser** tool. To switch to the rectangle mode, hold down the **Cmd** key.

Inserting Nodes

To insert a new node on a vector or curve with the Edit tool:

1. Press and hold the **CMD** and **OPTION** keys.
2. Position the mouse cursor on the point where you want to insert the node and click the mouse button. The mouse cursor will turn into the knife and the new node will appear in that place.

Using the "position" tool to insert nodes:

1. Activate the so-called position tool by pressing the mouse button **on a vector or curve** while the **CTRL** and **OPTION** keys are also pressed down. The mouse cursor will turn into a "target" with its coordinates shown.
2. Drag the mouse cursor along the primitive (it will stick) to "take aim" precisely and release the mouse cursor where you want the node to be inserted.

Using the Knife tool to insert nodes:

1. Activate the Knife tool .

2.1. Click on the point on the contour where you want to insert a node.

2.2. Press the mouse button anywhere on the empty area of the editing field and drag the mouse to form a "knife line". After you release the mouse button new nodes will be inserted at all points where this line crossed the outline. Hold down the **SHIFT** key to constrain the direction of the "knife line" to 15-degree increments.

If the "knife" line will cross two lines you may find a part of the glyph to "cut off". Hold the **OPTION** key to limit Knife tool to insert new nodes only.

Using the Add Corner, Add Curve and Add Tangent tools:

1. Activate one of the tools.
2. Click on any outline point. The Corner tool will add a straight line; the Curve tool will add a smooth connection and curve and the Tangent tool – a sharp connection and curve.

➲ **Note:** You can insert nodes on the closing vector that automatically connects the first and last nodes of a contour. If you insert nodes on the first half of a closing vector (closer to the ending node of a contour), then the new node will be added to the contour. If you insert the node on the last half of the closing vector, then it will be inserted before the starting point.

Meter Tool

With this tool you can measure any distance and angle in your glyph. It is very useful if you want to create very precise, extremely high quality symbols.

To measure distances between two points:

1. Press on the Tools toolbar. The **Meter** bar appears (if not yet) in the top part of the Glyph Window:

   This is a brief description of the fields on the Meter bar:

	Absolute position of the point (relative to the glyph zero point)
	Horizontal and vertical distance (from the beginning to the end of the metering line)
	Geometric distance
	Angle of the metering line.

   Note that you can switch the Meter bar on or off at any time if you click on the button in the bottom-left corner of the Glyph window.

2. Position the mouse cursor on your first point.

3. Press the mouse button and drag the mouse to the second point. In the Meter bar you will see the vertical, horizontal and direct distance between two points and the angle of the vector connecting these points. Hold down the **Shift** key while you drag the mouse to constrain the measurement to 15-degree increments.

While you are dragging the mouse you will see that the **Meter** cursor sticks to any object that it can find in the editing field.

To measure the distance from a contour:

1. Put the mouse cursor on the contour from which you want to measure.
2. Press the mouse button and drag the mouse to what you want to measure to. Hold down the **Shift** key and the direction of the mouse's movement will be constrained to the normal direction of the contour starting point.
3. When you're done, release the button.

With the Meter tool you can not only measure angles and distances but also mark glyph elements with guidelines.

Press the mouse button while holding down the **Cmd** or **Ctrl** key and measure the distance. When you release the button a popup menu appears. Select the **Add slanted guideline** command and a slanted guideline will be added along the meter tool's arrow. Note that the next guideline that you drag from the rulers will be parallel to this one.

Using the Drawing Tool

The easiest way to create a new contour is to use the Drawing tool: .

You can create a new contour or you can continue any existing contour. If you want to add new nodes to the existing contour, just activate its first or next node:

The last node of the open contour is activated

1. **To add a point**, just click the mouse button.

2. If you want **to create a line point**, just release the button. If you want **to define a curve**, drag the mouse to set the position of the curve control vector:

3. **To adjust the position of the curve control vector** without moving the control vector of the previous curve, hold **Option** and drag:

   You can press and release the **Option** key while you drag the mouse – when **Option** is released you are defining the positions of the control vector that belongs to the previous curve *and* the control vector of the next curve. When **Option** is pressed, you are not moving the previous curve's control vector.

4. When you are adding a new node, you can hold the **Cmd** key to not move the curve control vector but **move the node** itself.

5. Finally, you can use the **Shift** key at any time **to constrain the direction** of the line (if you are holding the **Cmd** key) or a curve control vector.

To close the contour, just click on its starting point and drag the mouse to set the direction of the control vectors.

Adding Points to a Contour

In addition to the Drawing tool you can use three more tools to create a new contour or to add points to an existing contour. These tools are: Add Corner, Add Curve and Add Tangent.

To create a new contour:

1. Activate one of the tools.
2. Click anywhere in the empty area of the glyph window to create the first point of a new contour. Drag the mouse to put new node into correct position.
3. Click again in the empty area to add a corner line, curve or smoothly connected curve (with Add Corner, Add Curve or Add Tangent tools respectively).
4. Continue the procedure until your newly defined contour is complete.
5. You can switch to the Drawing tool at any time and use it to add new points to a contour you are creating.

➲ **Note:** A new node is added to the contour if the last node of the contour is highlighted. If it is not highlighted a new contour is started.

To highlight a node click it. To deselect it, press the **Esc** key on the keyboard.

You can move outline nodes with the Add ... tools. Note that if you click on the contour (not on the node), a new node is inserted. The type of node depends on the tool you are using. To prevent adding a new node, hold down the **Option** key when you click on the contour.

Converting Primitives

Sometimes you may want to convert a curve to a vector or vice versa**. To convert a curve to a vector** "delete" (**Ctrl**-click while dragging) one of the control points of the curve, or "delete" the curve while you are in the non-node editing mode.

To convert a vector (normal or closing) to a curve drag an inside point of the vector while holding down the **Option** key or drag one of the nodes toward the middle of the vector holding down the **Option** key.

To convert a curve to a 1/4 part of an ellipse (the curve's control vectors will be treated as an ellipse axis), press the **Option** key and click on the curve.

You can also **convert** curves and vectors **with the popup menu**. **Ctrl**-click the end node of the segment and select the **Convert** command in the popup menu. With this command a line segment is converted to a Bezier curve, a Bezier curve to a TrueType curve and a TrueType curve to one or more Bezier curves.

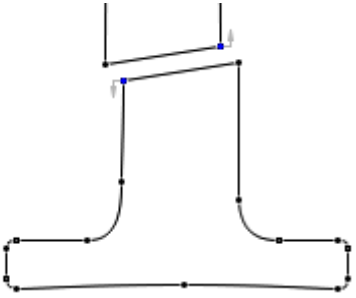
Breaking and Joining Outlines

To break the contour with the Edit tool hold down **Cmd** and **Option** and click the node where you want to break the contour.

To break the contour with the Knife tool just click on the node.

When a contour is broken its first and last nodes are highlighted by diagonal crosses:

You can use the Knife tool **to "cut out" part of the outline:**



1. Activate the Knife tool.
2. Press the mouse button and drag the cursor to define the "cutting line".
3. Release the button. Note that you can only cut part of a single contour, like in the sample picture above.

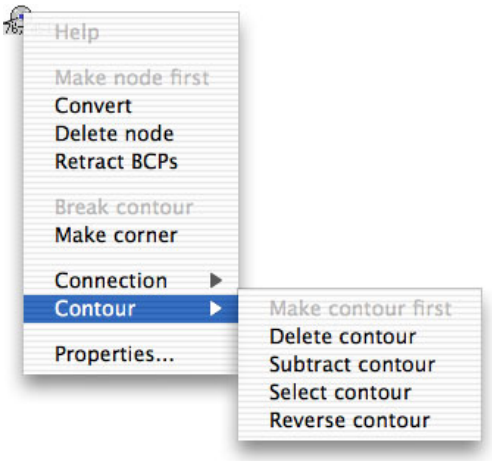
➲ **Note**: Deleting a selection with the **Edit>Delete** command breaks contours too.

To join two contours you need to move the starting or ending node of one contour to the starting or ending node of another contour.

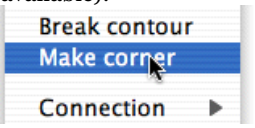
Hold down the **Option** key **to prevent the contours from joining**.

Node Commands

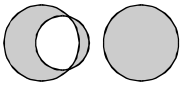
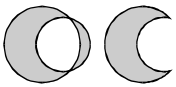
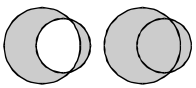
If you **Ctrl**-click a node you will see a popup menu with many useful commands:



Below is a description of all the commands in this menu:

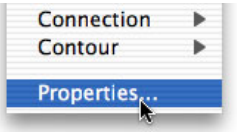
Make node first	Starts the current contour from the selected node (i.e. makes it the startpoint). This command is useful when you need to join contours since you can only connect starting and finishing nodes.
Convert	Cycles the node type from line to Bezier curve to TrueType curve.
Delete node	Removes the node.
Retract BCPs	Removes the control vectors of the node, making it sharp.
Break contour	Breaks the contour at the selected node.
Make corner	Makes a 90 degree corner (this operation is not always available): 
Connection	Popup menu with connection settings. You can use it instead of double-click or Option+double-click.
Contour	Set of commands related to the contour to which the selected node belongs (described below).
Properties	Opens the Node properties panel.

Contour commands:

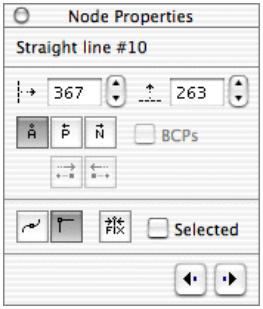
Make contour first	Begins the glyph from the selected contour. Renumbering contours is important for hint-substitution programming.
Delete contour	Removes the contour 
Subtract contour	"Subtracts contour" from the outline 
Select contour	Reverses the selection state of the contour.
Reverse contour	Reverses the contour direction: 

Node Properties

CMD-click the node or **CTRL**-click and choose the **Properties** command in the menu:



You will see the Node properties panel:



In this properties panel you can control the position of the node, the alignment type, the selection status of the node and the position of the control points of the curves.

To change the position of a node:

1. Select the origin point you want to use and set the coordinates of the node. By default the origin is the glyph's origin point . With the buttons you can select the previous or next node as the origin point.
2. Modify the coordinates of the node in the edit boxes. You can use the spin buttons to increase or decrease the coordinates. The new coordinates will be applied to the node when you press the **Enter** key on the keyboard or move the focus from one edit control to another or when you close the property panel by clicking on a free space in the edit field.

To change the connection mode for a node: select the new mode by clicking on one of the buttons in the lower part of the panel.

To change the selection state of a node: modify the state in the **Selected** check box.

To edit the position of the curve's control vectors: switch on the **BCPs** check box (it will be gray if you are editing a node between two vectors) and modify the relative position of the previous or next control point that belongs to that node.

Use the buttons with arrows **to edit the previous or next node**.

➲ **Tip:** when you are editing node positions in the Properties panel, press the **Enter** key to accept changes and move the focus to the editing field of the Glyph Window. There you can use the keyboard to move the selected node and the **Page Down**/**Page Up** keys to select another node for modification. You will see the node properties change in the Properties panel as you move the node by keyboard or mouse. Click **Option**+**Enter** to put the focus on the Properties panel to set the node position more precisely.

Selections

Many operations can be applied not only to single nodes or graphical primitives but also to several nodes together. For example, you may want to move many nodes or delete part of a contour. First, select the nodes that you want to process.

To select nodes with the selection rectangle:

1. Make sure that the Edit tool is active.
2. Press the mouse button anywhere in the empty area and drag the mouse to surround the nodes with a rectangle. Hold down the **Shift** key to reverse the selection state of the nodes.

To select or deselect individual nodes just shift-click them.

To select the contour segment (line or curve) – shift click on it.

To select the whole contour double click the contour (not a node) with the mouse cursor. To add another contour to the selection **Shift**-double click it. Alternatively you may press the **Cmd** key and click anywhere on the empty area close to the contour (the Magic Wand cursor appears). Hold down the **Cmd** and **Option** keys to select nested contours too.

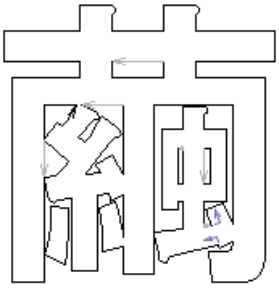
To select all the contours in a glyph use the **Edit>Select all** command.

To deselect all nodes click the mouse button somewhere in the free space of the editing field or use the **Edit>Deselect** command.

To reverse the selection state of all nodes in the glyph use the **Edit>Invert selection** command.

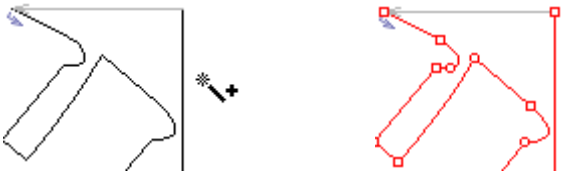
Using the Magic Wand Tool

With the **Magic Wand** tool you can easily and precisely select contours. It is especially useful when you are working with glyphs that have many contours, such as Far-Eastern ideographs:

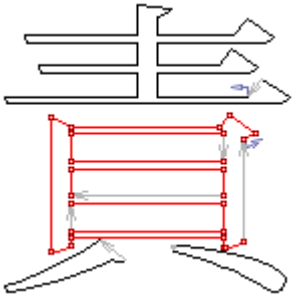


To select the contour with the Magic Wand tool just **CMD**-click anywhere near the contour. You don't need to be precise – TypeTool will automatically locate the closest contour.

To reverse the selection state of the contour, hold down the **SHIFT** key and **CMD**-click anywhere near it:



You can also select a contour and all contours that are inside it. Just hold the **OPTION** key when you using the Magic Wand tool:



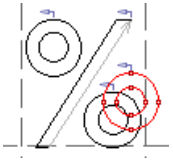
Moving the Selection

You can move the selected part of the contour by mouse – drag any selected part of the contour or use the arrow keys. If you press the arrow key then the selection will move in that direction by one font unit. Hold down **SHIFT** while pressing the arrow keys to accelerate the movement of the selection.

Copying the Selection

Sometimes you need to copy glyphs or parts of glyphs to another place in the font or even into a different font. With TypeTool you can put any part of a glyph or an entire glyph (with hints, guides, etc.) into the Clipboard and paste it into a different place.

To copy parts of the glyph's outline use the commands from the **Edit** menu or the buttons on the Standard toolbar:

Cut		To copy a selected part of the glyph onto the Clipboard and delete it from the glyph.
Copy		To copy a selected part of the glyph onto the Clipboard.
Paste		To add a contour part copied to the Clipboard into the current glyph as a new contour
Insert		To replace the current selection with the Clipboard contents.
Delete		To remove the selected part of a glyph's contour.
Duplicate		To insert a copy of the selection into the current glyph as a new contour 

When you use the **Paste** command, the selection is always pasted without a shift. When you use the **Duplicate** command, the copy is always pasted with an offset 100x100.

Because the Clipboard is used as a buffer for copying contours you can paste glyphs and their parts not only to the current font but to any glyph of any font of any application that is compatible with TypeTool (ScanFont 3.0, for example).

Copying an Outline to Other Applications

With TypeTool you can exchange outline data with other vector-editing programs. Supported programs are Adobe Illustrator and Macromedia FreeHand.

To copy part of the glyph's outline to a vector-editing program use the usual copy procedure. The selected portion of the outline will be copied to the Clipboard. Then switch to your vector-editing program and select the **Paste** command from the **Edit** menu.

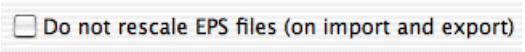
You can change the size of the copied outline using the scale transformation that is supported by your vector-editing application.

To copy an outline from a vector-editing program into TypeTool select the outline object that you want to copy and choose the **Copy** command from the **Edit** menu (in the source application). To place the copied outline in TypeTool switch to TypeTool (Glyph Window) and select the **Paste** command from the **Edit** menu.

Notes:

1. TypeTool can only edit font outlines, not features such as color of outline, outline width or fill color. Regardless of the settings you have in the vector-editing application, only information about outlines will be copied to TypeTool.
2. To see how an outline will look in TypeTool before you copy it, in the source vector-editing application select the object and set the fill color to 100% black, and the width of the outline to none.

When you paste an outline from another application into TypeTool it scales it to fit the font's UPM. To prevent the outline from automatically scaling, use the **Do not rescale EPS files** option located at the top of the **Preferences>General** dialog box:



Transform Selection

Sometimes you need to scale, rotate or slant a whole or part of a glyph outline. In TypeTool you can do this using different methods:

1. Using the Transformation panel (described below)
2. Using the basic transform tools
3. Using the Transform Selection operation

Using Basic Transform Tools

You have access to three basic transform tools:

	Rotate	Rotates the contour
	Scale	Scales the contour
	Slant	Slants the contour vertically or horizontally.

To transform the outline:

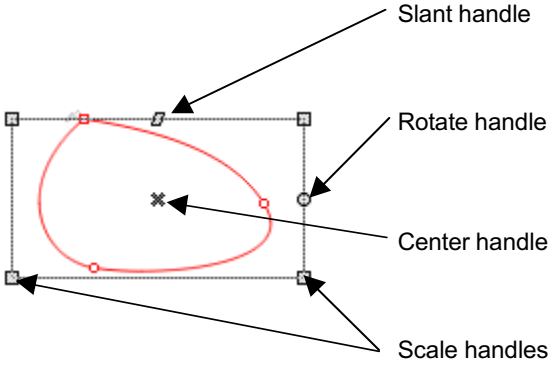
1. Select part of the outline you want to transform or undo all selections to transform the entire glyph outline.
2. Activate one of the transform tools.
3. Position the mouse cursor at the center of transformation, press the mouse button and drag the mouse to make the transformation. Remember that you can press the ‘`’ key at any time to get a high-quality preview of the transformed outline.
4. Use the **Shift** and **Cmd-Shift** keys to constrain the transformation.
5. Release the mouse button to complete the transformation of the outline.

Using the Transform Selection Operation

To activate the Transform operation click the **Transform Selection** button in the Tools toolbar or **Ctrl**-click the selected portion of a glyph and select **Transform** in the popup menu.

Or you can double-click on any selected graphical primitive to activate the **Transform** operation.

When this operation is activated, you will see a transformation rectangle surrounding the selected area. If nothing is selected, the entire glyph will be subject to transformation.



So, what do all these handles mean, and how can they be used?

To move a selection:

1. Position the mouse cursor somewhere inside the transformation rectangle but not on the center handle.
2. Press the mouse button and drag the rectangle to its new place.
3. Release the button. The selection will be moved.

To scale or skew a selection:

1. Position the mouse cursor on one of the scale handles ▣.
2. Press the button and drag the mouse. You will see that the transformation rectangle is scaled. Hold down the **Shift** key on the keyboard to constrain the scale proportionally.
3. Release the button when you are done. The selection will be modified.

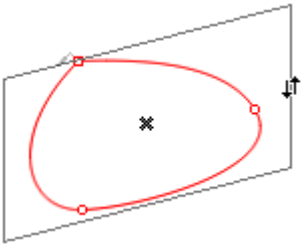
To rotate a selection:

1. Move the mouse cursor onto the rotation handle ○.
2. Press the mouse button and drag the mouse. The transformation rectangle will rotate around its center. Hold down the **Shift** key to constrain the rotation angle to 15-degree increments. You can also use the slant handle for rotation – just press the **Option** key to alternate between rotate or slant.
3. Release the button to accept the rotation.

To move the center of rotation just drag the center handle ✖ by the mouse to its new position.

To slant a selection:

1. Move the mouse cursor onto the slant handle .
2. Press the mouse button and drag the mouse. The transformation rectangle will be slanted. Hold down the **Shift** key to constrain the rotation angle to 15-degree increments.
3. To slant in vertical direction, hold the **Option** key and drag the rotate handle:



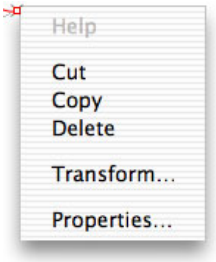
4. Release the button to accept the slanting.

Double-click in the editing field or press **Enter** to accept the completed transformation or press the **Esc** key to reject it.

You can use the arrow keys while the Transform tool is active to move the selection by one font unit in the direction of the key you clicked. **Shift+arrow** keys move the selection by 10 font units at each key click.

Selection Commands

When a part of the glyph is selected, **Ctrl**-click it to get access to the popup menu:



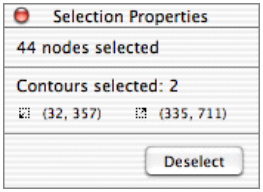
These commands are just copies of the **Edit** menu commands:

Cut	Copies the selection to the Clipboard and removes it.
Copy	Copies the selection to the Clipboard and leaves original untouched.
Delete	Removes the selection.
Transform	Activates the Transform operation.
Properties	Opens the Selection properties panel.

Selection Properties Panel

To make the Selection Properties panel visible, choose the **Properties** command in the selection context popup menu or use the **Option-Return** keyboard shortcut.

The Selection Properties panel is very simple:

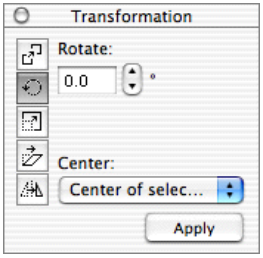


It contains the following information about the selection: the number of selected nodes, the number of selected contours, and the selection bounding box's bottom-left and top-right corner coordinates.

You can click on the **Deselect** button to discard the selection and get the Glyph Properties panel instead.

Transformation Panel

The Transformation panel allows you to apply several simple transformations to the selected area or to the whole glyph. To open the Transformation panel you can select the **Transformation Panel** command in the **Window** menu:



To transform the glyph or the selected area:

Select the type of the transformation by clicking on one of the buttons in the left and the transformation options in the right area:

Shift	Shift (x, y): 0 0 u	Enter a distance to move the selection
Rotate **Slant**	Rotate: 0 ° Center: Zero point	Enter the rotation or slant angle (degrees, clockwise) and select a center of rotation.
Scale	Scale (x, y): 100 100 % Uniform Center: Zero point	Enter the scaling factor and select a center point of transformation. Use the Uniform option to scale proportionally.
Mirror	Mirror axis: 0.0 ° Center: Zero point	Enter the direction of the mirror axis and select the center point of the transformation. Use the buttons to mirror horizontally or vertically quickly.

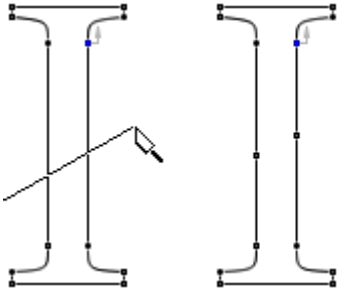
Press the **Apply** button to apply the transformation to the selected area.

Building an Outline From Blocks

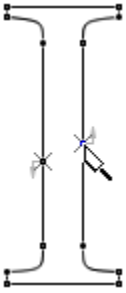
Now you know how to select parts of an outline and copy it, so let's do a few experiments to show how to use this knowledge.

Suppose that we have an 'I' character and we want to create an 'H' character.

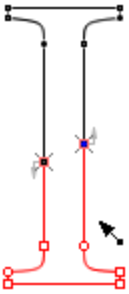
1. Open the 'I' in the Glyph Window (double-click the 'I' cell in the Font Window).
2. Cut the character in the middle. Activate the Knife tool, press the mouse button at the left of the character, hold the **Option** key on the keyboard and drag the mouse cursor to the right to define a cutting line. Release the mouse button:



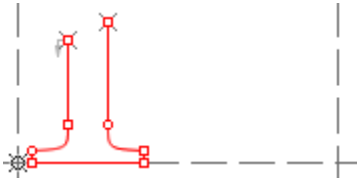
3. Click on each inserted node to break the contour:



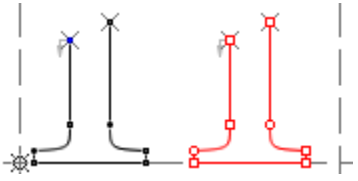
4. Select the bottom half of the ‘I’. Temporarily activate the Edit tool (“click” on the **CMD** key on the keyboard) and **CMD**-click near the bottom area of the glyph:



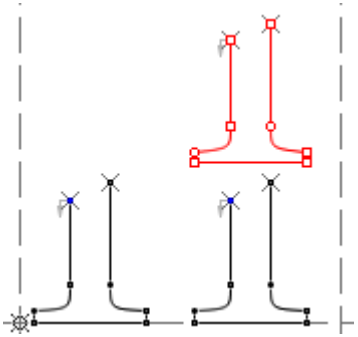
5. Copy it to the Clipboard with the **Edit>Copy** command.
6. Go to the ‘H’ character. Use the **Glyph>Reset** command to remove the contents of all editing layers.
7. Use the **Edit>Paste** command to place a copy of the ‘I’ bottom:



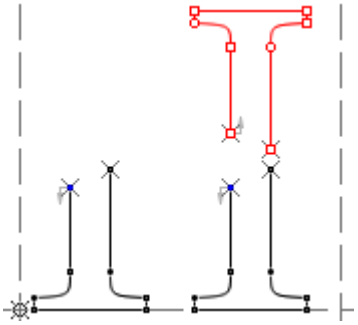
8. Use the **Edit>Paste** command to make a second copy. Shift-drag the second copy horizontally:



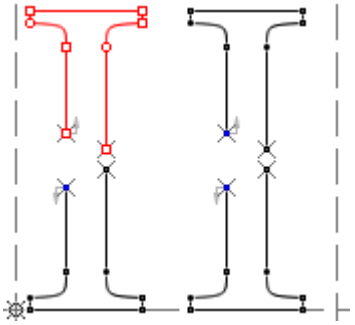
9. Copy and paste it again and shift-drag it to a place above the second segment:



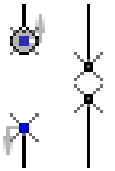
10. Use the Rotate (180°) operation in the **Transformation** panel to flip the selected segment:



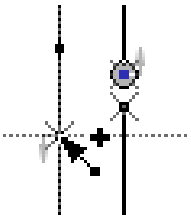
11. Duplicate the top segment and locate the copy above the bottom-right segment.



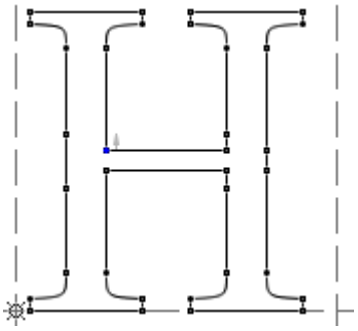
12. Click on an empty area of the Glyph window and then click on the top-left node of the top-left segment to highlight it:



13. Activate the **Add Corner** tool click somewhere and drag the line to connect the left line of the top-left and bottom-left segments:



14. Then click on any other starting or ending node of the contour segments and use the **Add Corner** tool to connect them



You'll notice it takes more time to read the instructions than to actually perform the procedure.

Outline Actions

You can find the actions described below in the **Tools** menu and in the Glyph Window's context menu:

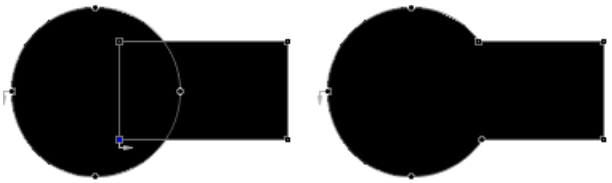
Check Connections	Analyses an outline and detects the types of connections between outline primitives (lines and curves).
Reverse All Contours	Reverses the direction of all contours of the glyph. This action can be useful if you want to completely convert a Type 1 glyph to a TrueType glyph. In TrueType and Type 1 format the default direction of the contours are opposite.
Nodes at extremes	Automatically inserts nodes at the extreme points of curves. We highly recommend that all curves have nodes at their extreme points.
Snap to Guides	Move all nodes on to guidelines, hints or grid if they are sufficiently close to them. This action will "snap" nodes only to the guiding layers that are currently visible.
Merge Contours	Combines all overlapping parts of the outline. This operation and the two following operations are applied to all contours that have at least one node selected. If nothing is selected, they will be applied to the whole glyph.
Get Intersection	Leaves only areas that are covered by at least two contours.
To TrueType Curves	Converts all Type 1 (3^{rd}-order) curves to TrueType (2^{nd}-order) curves.
To Type 1 Curves	Converts all TrueType (2^{nd}-order) curves to Type 1 (3^{rd}-order) curves.

Merging and Intersecting Contours

With the **Merge contours** and **Get intersection** commands, which are available in the **Tools** menu, you can perform very interesting operations on contours.

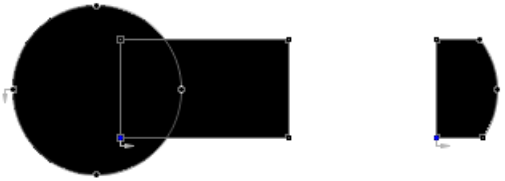
Both operations are applied to contours that have at least one node selected or to the whole glyph outline if nothing is selected.

The **Merge contours** command combines contours, removing all outline overlapping and keeping the filled result unchanged:



The **Merge contours** command is the outline equivalent of the Boolean "OR" operation.

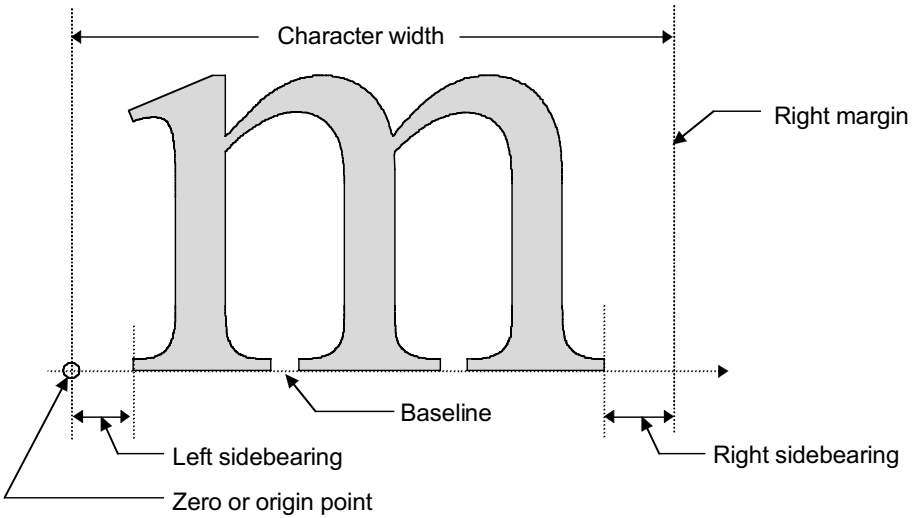
The **Get intersection** command will keep only the area of intersection, removing all other parts:



This command is the outline equivalent of the Boolean "AND" operation.

Metrics

The Metric data of a glyph includes information about the horizontal and vertical width. Glyphs have an *origin point*, a *baseline*, *sidebearings*, and *left and right margins*:



The baseline is used to align characters in a series. The left and right margins are used to define the positions of sequential characters in a series when the horizontal writing mode is selected. In the vertical writing mode the left and right margins are used to horizontally align characters and the top margin is used to vertically align characters.

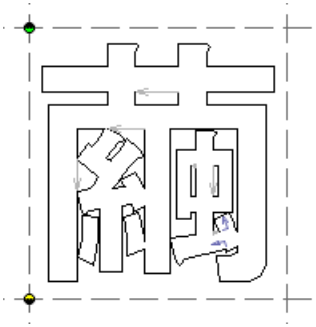
In TypeTool, the position of the origin point is the position of the left margin in the horizontal direction and the position of the baseline in the vertical direction. However, you can modify the position of any of the four margins. If you move the baseline or left margin line you will shift the entire glyph, but as a result the left margin is moved.

Editing Metrics

TypeTool has a special window for editing glyph metrics, of course, but you can make small adjustments right in the Glyph Window, using the main Edit tool.

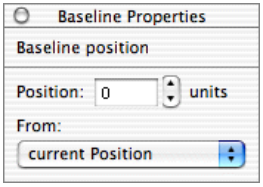
To change glyph metrics use the mouse and drag the left or right sidebearing or the baseline.

In TypeTool you can define a vertical advanced vector for the glyph (this information is used in fonts which may type vertically, from top to bottom). To define a vertical advanced vector, press and hold the **SHIFT** key before moving the base line:



Baseline Properties Panel

With this property panel you can adjust the position of the glyph's baseline. To open it **Ctrl**-click on the baseline and select the **Properties** command in the popup menu or just click the baseline while holding down the **Cmd** key on the keyboard.

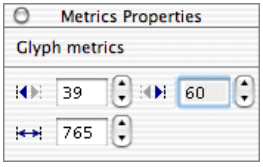


To change the position of the baseline:

1. Select the base level of the modification. It can be the old position (for relative offset) the top of the glyph, or the bottom of the glyph.
2. Change the position of the baseline relative to the base level.
3. Press the **Enter** key or click anywhere in the editing field to apply the changes.

Metrics Properties Panel

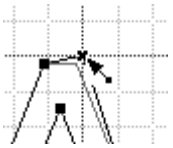
To open the metrics property panel position the mouse cursor on the left or right glyph margin, **Ctrl**-click and select the **Properties** command, or just **Cmd**-click the mouse button on one of the margins.



In this panel you can modify a glyph's sidebearings and/or width.

Grid Layer

This layer is very simple: if the Grid is on, you will see a grid of vertical and horizontal lines in the Glyph Window. If snap-to-grid is on (which it is by default) any node that you move will snap to the gridlines.



You can adjust the grid frequency on the Glyph Window page of the Preferences dialog box (**Application>Preferences** command):

Guidelines Layer

Guidelines are straight lines that are used to guide the drawing of specific elements of a glyph. Guidelines can be vertical, horizontal or slanted.

Guidelines can be slanted at any angle from -45 to +45 from the vertical or horizontal direction. Slanted guidelines can help to mark *italic* characters, or specific slanted elements in normal characters, like the inner bar in the letter 'N'.

You can see little numbers giving the position and slanting angle of each guideline near the edges of the editing field of the Glyph Window where the guidelines cross the rulers.

There are *local* and *global* guidelines. Local guidelines appear only in the glyph where they were set. Global guidelines appear in all glyphs of the font. Global guidelines are very useful to mark important levels in the font (by using horizontal global guidelines) or to set the base direction of an italic or oblique font (using slanted vertical guidelines).

Editing Guidelines

Be sure that the Edit tool is active and the guidelines layer is visible – use the **View>Show layers** menu to switch it on. Note that the Guidelines layer will automatically switch on if you add a new guideline.

To add a new guideline:

1. Position the mouse cursor on the horizontal ruler bar (for a horizontal guideline) or on the vertical bar (for a vertical guideline).
2. Press the mouse button. The bar will appear "pressed" and the new guideline will appear.
3. Holding the mouse button down, drag the guideline to the desired place and release the button.

To move the guideline:

1. Move the mouse cursor onto the guideline that you want to move. Be sure that no other objects (such as nodes or hints) are near the cursor.
2. Press the mouse button and drag the guideline to the new place.

To slant the guideline:

1. Move the cursor onto the guideline near one of the sides of the editing field of the Glyph Window.
2. Press the mouse button. The mouse cursor will change to a pair of curved arrows that shows you the guideline slant direction.
3. Moving the mouse, slant the guideline to the angle that you want. Hold down the **Shift** key to constrain the slanting angle to 3-degree increments.

To remove the guideline:

I.1. Start moving or slanting the guideline.

I.2. While holding down the mouse button, press the **Ctrl** key and then click the mouse button again.

II.1. Position the cursor on the guideline and **Ctrl**-click the mouse button.

II.2. In the menu, select the **Delete** command.

You can also remove any guideline or hint by simply dragging it from the editing field of the Glyph Window.

To remove all guidelines use the **Remove guidelines** command in the **Tools>Hints & Guides** menu. Options of this command include:

Both	to remove all guidelines
Vertical	to remove only vertical guidelines
Horizontal	to remove only horizontal guidelines.

The same commands are available in the rulers popup menu that appears if you **Ctrl**-click the vertical or horizontal ruler.

➲ **Note:** While you are editing the guideline, its parameters are shown on the status bar.

Guidelines Popup Menu

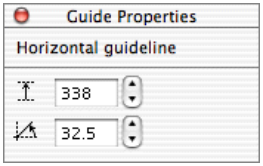
More commands are available in the guideline's popup menu.

There are two different guideline popup menus – one for local and one for global guidelines. Both menus include **Align**, **Delete** and **Properties** commands. The **Properties** command, as usual, will open the Property panel for the active guideline. The **Delete** command will remove the active guideline. The **Align** command is available only for slanted guidelines and will align them to the vertical or horizontal axis (i.e. remove their slant and make them vertical or horizontal guidelines).

Global guidelines have an additional set of commands located in the **Guideline Is** submenu. By selecting the commands in this submenu you can set the font parameters: **Ascender**, **Descender**, **Caps Height**, **x Height** or **Visual Ascender** and **Descender** that are used to set 100% zoom in the Glyph Window.

Guidelines Properties Panel

To open the guideline properties panel, **Cmd**-click the guideline or **Ctrl**-click it and choose the **Properties** command from the popup menu:



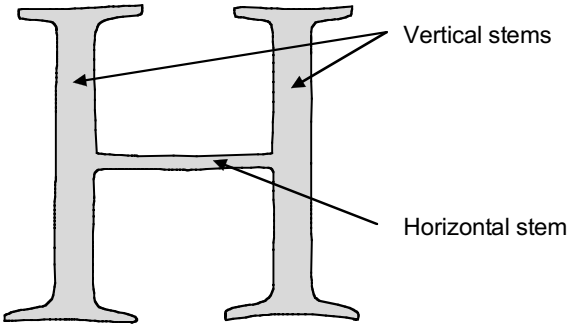
In this properties panel you can change the position and slant angle of a guideline. To apply the changes press the **Enter** key or just click anywhere outside the property panel.

Hints Layer

Hints are used by the font rasterizer to improve a glyph's appearance on devices with low output resolution, like computer monitors or low-res printers.

There are two hinting methods applied to Type 1 fonts (hints for TrueType fonts are always generated automatically): font-level hinting and character-level hinting. Font-level hinting is generated automatically in TypeTool, so you don't have to edit it manually. In the Glyph Window you can see Type 1 character-level hints.

Character-level hinting is applied to the characters' stems:



All important stems in a glyph should have stem hints, a pair of vertical or horizontal lines. The information in the hint includes not just the position of each of the two lines that "build" the hint, but also the position of one (major) line and the width of the hint.

You can declare stem hints in TypeTool just by dragging them and modifying their width. Because hints in TypeTool are very "intelligent," they automatically snap to the contour, minimising your work. In most cases the autohinting algorithm that is included in TypeTool produces good results – usually not any worse than the results of manual hinting.

Editing Hints

Editing hints is very similar to editing guidelines. You can add new hints through the ruler bar of the Glyph Window; drag them with the mouse; and delete them by using the menu command or by **Ctrl**-clicking when dragging.

In contrast to guides, hints consist of two lines that can be moved together or separately. Hints cannot be slanted.

To add a new hint:

1. Position the mouse cursor on the horizontal ruler bar (for a horizontal hint) or on the vertical bar (for a vertical hint).
2. Press and hold the **Cmd** key. Press the mouse button. The bar will appear "pressed" and a new hint will appear. Release the **Cmd** key.
3. Holding the mouse button, drag the hint to the desired place and release the button.

To move a hint:

1. Move the mouse cursor onto one of the hint's lines.
2. Press the mouse button and drag the hint to its new place. Both hint lines will move together.

To move a hint's lines separately hold down the **Shift** key while dragging one of the hint's lines. Using this procedure you can change the width of the hint.

➲ **Note:** While you are editing the hint, its parameters are shown on the status bar.

To remove a hint:

I.1. Start editing the hint.

I.2. While holding the mouse button down, press the **Ctrl** key and then click the mouse button.

II.1. Position the cursor on the hint and **Ctrl**-click the mouse button.

II.2. Select the **Delete** command from the popup menu.

Hint Popup Menu

To open the hint popup menu, CTRL-click one of the hint lines.

The **Hint** popup menu includes the following commands:

Reverse	Reverses the direction of the hint
Delete	Removes the hint
Properties	Opens the hint property panel.

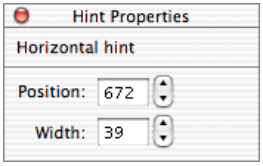
Hint Commands

The **Tools>Hints & Guides** menu contains several commands related to hints:

Remove Hints	Removes vertical or horizontal or all hints. This command is duplicated in the rulers context menu.
Autohinting	Automatically generates stem hints for the current glyph.

Hint Properties Panel

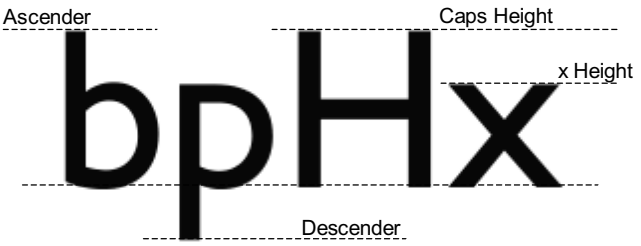
To open the hint property panel, CMD-click one of the hint lines or CTRL-click and choose the **Properties** command in the popup menu:



In the hint properties panel, you can modify the position of a hint in the upper edit box and modify the width of the hint in the lower box. Press the ENTER key or click the mouse outside of the properties panel to apply the changes.

Vertical Metrics

Every font has several vertical metrics values for alignment of text:



The **Ascender** line defines the position of the top of lowercase characters (usually the topmost point of the Latin 'b').

The **Descender** line defines the position of the bottom of the lowercase characters (usually the bottom point of 'p').

The **Caps height** defines the height of the uppercase characters (without overshoot); usually it is the height of 'H'.

The **x Height** is the height of most lowercase characters, like 'x' or 'v'.

In TypeTool you can modify the vertical metrics values in the Font Info dialog box, but you can also preview and change them visually in the Glyph Window.

Make sure that the Vertical metrics layer is active.

In the Editing field vertical metrics appear as gray lines with a label at the left:

Ascender

Descender

Caps height

x Height

To change a metric, just drag its line with the Edit tool.

VectorPaint Tools

VectorPaint is TypeTool's unique set of tools that allow you to paint vector contours but look and feel like bitmap tools. You can choose brushes, pens and even enter text. The idea of VectorPaint is that all the tools produce contours that combine with the existing glyph contours using our unique contour-processing technology.

All VectorPaint tools are located in the Tools toolbar:

The type of interaction between existing and new contours depends on the selected color mode. This process is very fast and is completely transparent to you, so if you switch on the preview mode (where the glyph appears filled), the illusion of bitmap-like editing of a contour-based glyph image is very realistic.

All the tools can work in 4 different color modes:

Transparent	Newly created vector objects that are generated by the application of VectorPaint tools do not interact with the existing glyph's contour and appear selected for easy editing.
Automatic	The color of the brush depends on the point where you begin drawing. If you begin in a white area, a white brush will be selected, if in black, a black brush will be selected. Use this color mode to easily extend white or black areas of the glyph.
Black	Generated contours are added to existing contours, expanding the black area of the glyph. It looks like a black brush applied to a black picture.
White	New contours are subtracted from existing contours, simulating a white brush or eraser.

Here is an example of a brush stroke applied with Transparent, Black and White “colors”:

Here is a list of all available VectorPaint tools with a short description of each:

	Pen (Contour)	Used to create new contours or modify existing ones. It is not really a “paint” tool, because it deals with contours, but it is a very natural and flexible tool used to adjust the result of the application of VectorPaint tools.
	Brush	Exactly that – a brush. It can be round or calligraphic. A calligraphic brush can be of any size and slant angle.
	Line	Used to draw straight lines with a selected brush.
	Text	Used to enter text (vector based) using any TrueType font installed in the system.
	Ellipse	Used to draw ellipses or circles.
	Rectangle	Used to draw rectangles or squares.

Pen (Contour) Tool

With the Contour tool you can create new contours or modify existing contours in a more artistic manner than with the Edit tool. When you use the Contour tool, you can draw new contours just as you do on paper. TypeTool will trace your drawing and replace it with a series of curves and lines.

How to create a new contour

If you begin a contour in a free area (where the cursor has its ordinary shape), you will define a new contour. If you want to begin a new contour but its starting point is on an existing contour press the **CMD** key to force TypeTool to create a new contour.

How to modify an existing contour

When you move the cursor of the Contour tool onto an existing contour or node, it changes. If you begin drawing (without holding down the **CMD** key) the new contour will be inserted into the existing one. If the finishing point of your drawing is on an existing contour also, and the starting and finishing points are on the same contour, then the new drawing will replace the part of the existing contour that lies between the starting and finishing points.

How to draw a single curve

Hold the **SHIFT** key down when you release the mouse button after drawing a new line. Your drawing will be approximated by a single curve. This is a good way to draw a new contour step-by-step.

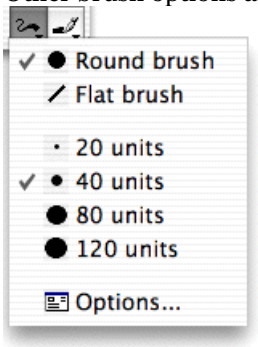
Brush Tool

The brush tool works like the usual bitmap brush that you find in any bitmap-editing program. You begin a brush stroke by pressing the mouse button. Draw the stroke by dragging the mouse and finish drawing it by releasing the mouse button.

To change the color of the brush, use the color selection buttons on the Tools toolbar:

	for the “empty” color
	for the “auto” color
	for the “black” color
	for the “white” color

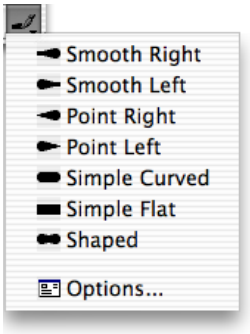
Other brush options are accessible in the VectorPaint options menu:



You can paint with round or calligraphic brushes of different widths:

	Round brush
	Calligraphic-style brush
	20 unit wide line
	40 unit wide line
	80 unit wide line
	120 unit wide line

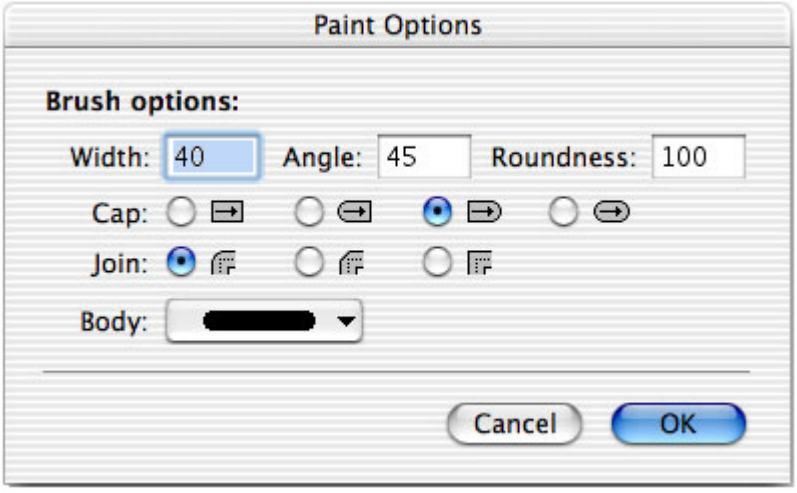
You can also specify a brush stroke shape. Press the button and select a shape in the popup menu:



This is an example of different brush strokes:

VectorPaint Options

You can also change the brush properties in the Paint Options dialog box. **To open the Paint Options dialog box**, choose the **Options** command in the brush options menu ( ). You will see following dialog box:



In the **Paint Options** dialog box you can enter the width of the brush and change the slant angle of a calligraphic brush. Additionally, you can select how the brush strokes are started and finished. Choose the brush's starting and finishing shape by activating one of the radio buttons.

You can also select the style of the connection between two sequential segments of brush strokes. It can be sharp, smooth or flat. Select one in the **Paint Options** dialog box. Icons near the radio buttons give an explanation of the styles of connections.

The **Body** popup menu in the Brush options area lets you to select the shape of the brush stroke.

Line Tool

The Line tool allows you to apply brushes to straight-line segments.

To draw a line segment:

1. Position the mouse cursor on the beginning point and press the mouse button.
2. Move the mouse to the end point and release the button. Hold down the **SHIFT** key to constrain the direction of the line to 15-degree increments.

Ellipse and Rectangle Tools

The Ellipse and Rectangle tools are very similar. The only difference is in the result.

To draw an ellipse or rectangle:

1. Select the tool that you want to use (or).
2. Position the mouse cursor on the spot where you want to place one of the rectangle corners (or on one of the corners of the rectangle that surrounds the ellipse).
3. Press the mouse button and drag the mouse to define the rectangle (or ellipse).
4. Hold down the **SHIFT** key to constrain the direction of the rectangle's diagonal to 15-degree increments.
5. Release the button to finish creating the rectangle (ellipse).

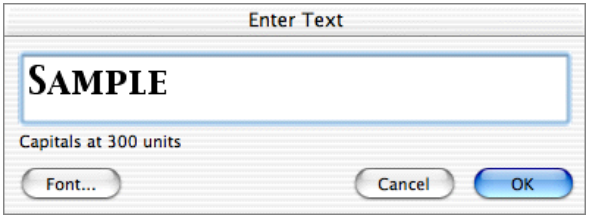
Text Tool

With the Text tool you can add text to a glyph. Carefully select the color mode when planning to use the text tool. It is usually best to use the Transparent painting color because in that mode the text stroke will not interact with the existing contour and you will be able to adjust its position using the Edit tool or Transform operation.

Select the **Automatic Activation of Transform Operation** option in the Paint Options dialog box. With this option on the Transform operation will be activated immediately after entering the text string, allowing you to modify its size or position.

To enter a string of text:

1. Select the text tool T in the Tools toolbar.
2. Position the mouse cursor (with the crosshair and the "suggested rectangle" of the future string) on the place in the editing field where you want to add the string.
3. Click the mouse button.
4. In the dialog box, enter the character string. Use the **Font** button to select the font that will be used:



   Below the sample string you will see the name of the current font and the size of the text string. The size is presented in font units. You can change the string size in the Font dialog box. The size of the placed text will be 10 times the selected point size. For instance, if you select a 24 pt. font you will get a string that will be 240 units in height.

5. Press **OK** to enter the string or **Cancel** to abort this operation.

Working with Composite Glyphs

Composite glyphs are glyphs made up of two or more components, like a letter plus an accent. One or more of the components are referenced. I.e. their contours are not actually present in the composite glyph, but are "copied" from and linked to some other character. Thus whenever the original component contour is changed all the composite glyphs that copy the component also change. The contour of composite components appears in dashed lines in the glyph window.

Composites have the advantage of allowing the user to create only one instance of a component that is frequently found in a font and reusing it without having to redraw it each time. Further if the design of the component changes it need only be altered once – in the original component. And finally, a composite takes up less room in the font than an outline, allowing for smaller font files.

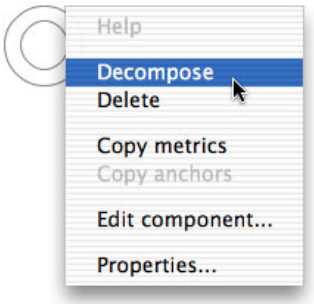
There are three operations related to composite glyphs: adding a component to glyphs, decomposing a component and positioning a component.

In TypeTool 2.0 you cannot create composite glyphs or add components to glyphs. But you can decompose a component or position a component in the composite glyphs of imported font.

Decomposing

To decompose a composite glyph select the **Decompose** command from the **Glyph** menu or from the Glyph Window default popup menu. The outlines of all components will be scaled and shifted according to their settings and added to the composite glyph. If the component glyphs had hints then these hints will also be added and a hint replacement program will be automatically generated. The link to the original component will be lost.

To decompose an individual component in a composite glyph **Ctrl**-click the component and select Decompose in the context menu:



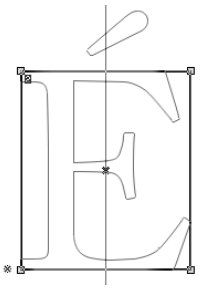
Component Positioning

To activate the component positioning operation, activate the Edit tool, position the mouse cursor on the component's outline and click the mouse button.

Alternately, if the current glyph is composite-only (so if doesn't have any "normal" outlines), use the **Page Up** and **Page Down** keys **to select a component for editing**.

You will see a control box surrounding the component with four corner handles, a cross in the center, a centerline and the number of the component in the components list.

To select another component, use the **Page Up** and **Page Down** keys or the **Tab** key.



To move the component position the mouse cursor inside the control box, press the mouse button and drag the control box to a new location. If you position the cursor on the cross in the middle of the control box you can set the position of the component more precisely because the cross will snap to the guiding elements while moving. Pressing the **Shift** key will constrain the movement to the horizontal and vertical directions, pressing **Cmd-Shift** allows the component to move only horizontally.

When you move the component, it will "stick" to its original position. You must hold down the **Option** key to avoid this.

You can also **use the keyboard to move the component**. Arrow keys move the component in one font unit increments and the **Shift**+arrow keys increase movement to 10 units.

To scale a component position the mouse cursor on one of the handles, press the mouse button and drag the mouse to change the size of the component. Hold the **Shift** key down to constrain the proportions of the component.

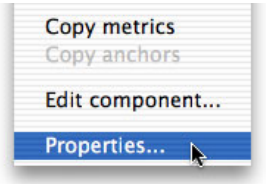
Just click outside the component's control box to accept the changes (you can also press the **Return** or **Enter** key on the keyboard) or press the **Esc** key to reject the changes. You can also click another component to select it for editing.

Some other useful commands are available in the popup menu that appears if you **Ctrl**-click the editing area while component tool is active:

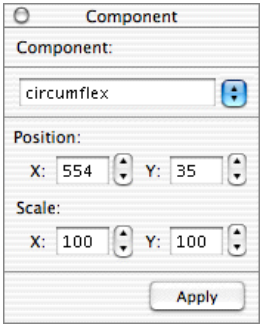
Decompose	Decomposes (adds the outline to the composite glyph) the current component.
Delete	Removes the component.
Reset Shift	Moves the component to the position (0, 0).
Set Scale 100%	Sets the scale at 100% for the component. Note that the Type 1 font format does not support scaled components.
H Mirror **V Mirror**	Mirrors the component (sets the scale for x or y direction to −100% and adjusts shift accordingly).
Edit Component	Opens a new Glyph Window with the currently active component.
Properties	Opens the Component Properties panel.

Component Properties

You can **set the precise size and position of the component**. Ctrl-click the component with the Edit tool. You will see a popup menu:



Select the **Properties** command in this menu and you will see the component Properties dialog box:



In this dialog box you can select a different glyph to be used as a component and set the component's position and scale. The component position is the distance between the composite zero point and the component's zero point.

➲ **Tip**: You may just double-click on the component to get the component Properties dialog box.

Importing and Exporting Glyphs

The most common format for contour-based data is Encapsulated PostScript (EPS). Because this format is a native format of a well-known contour-manipulation program, Adobe Illustrator, another name for it is Adobe Illustrator format (AI). If we go deeper, we see that EPS is much more common than AI, but due to a well-defined internal structure AI format has become a standard way of exchanging contour-based graphics.

With TypeTool you can export any glyph as an AI-compatible file and use it in any other program and you can import any EPS/AI file (compatible with Adobe Illustrator) to add to any glyph of your font.

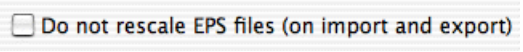
To export a glyph as an EPS/AI file:

1. Select the **Export to EPS** command from the **Glyph** menu.
2. Select the export folder and enter the name of the EPS/AI file in the standard File Save dialog box.
3. Press the **Save** button in the dialog box, and the EPS/AI file will be exported to the folder chosen.

To add an EPS/AI file to the current glyph:

1. Select the **Import from EPS** command from the **Glyph** menu.
2. Choose the file that you want to import in the standard File Open dialog box.
3. Press the **Open** button in the File Open dialog box, and the EPS/AI file will be imported, added to the current glyph, and selected to simplify its modifications.

When you import or paste an outline from another application into TypeTool it scales it to fit the font's UPM. To prevent an outline from scaling, use the **Do not rescale EPS files** option located in the **Preferences>General** dialog box:



You can also **export several glyphs at once**: switch to the Font Window (for example, by selecting the **Open Font Window** command from the Glyph Window's popup menu), select the glyphs that you want to export and select the **Export to EPS** command from the **Glyph** menu. You will see a Save File dialog box where you enter a prefix file name for the exported glyphs. Each glyph will be exported to its own file with the file name consisting of the prefix plus the sequential number of the exported glyph.

Printing a Glyph

To print a sample of the current glyph select the **Print** command in the **File** menu while the Glyph Window is active.

You will see a standard Print dialog where you can choose a printer and select other printing options.

To begin printing click the **Print** button. When printing is complete you will have a sample of the current glyph as seen in the Glyph Window. If you see nodes on the screen they will appear on the paper. If you are in preview mode the printed glyph sample will be filled.

Zoom mode on the screen will also be applied to the printed sample of the glyph. Using this feature you can print very detailed samples of the glyph.

Editing Font Metrics

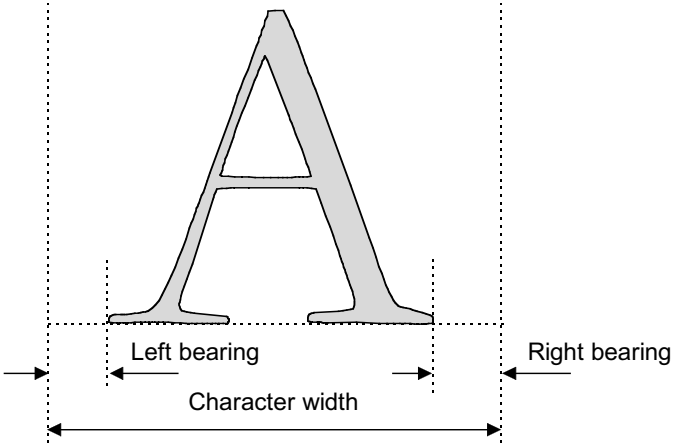
The tools in TypeTool that you can use to edit metrics data are common to all FontLab applications, so if you have learned how to use these tools in TypeTool you will be ready to use these same tools in any of the FontLab programs.

5

What are Font Metrics?

A program that aligns and spaces text calculates the total width of all the characters in a paragraph. It then adjusts the widths of the space characters that separate the words and tries to put as many characters as possible into one line. The information about the words that are used to make a paragraph, and the information about the width of the individual characters is the only information necessary. This is what is known as font metrics.

To position the individual characters, the page layout program must have information about each character's sidebearings:



Using this information the page layout program positions all the characters. Kerning information is also necessary in order to get the best results.

Kerning

Kerning information is used to adjust the space between specific pairs of characters. As you can see in the following picture some characters may be well spaced with just the bearings rectangle but other characters are not. To fix this problem a special technique called kerning has been developed.

Kerning defines the modification of inter-character space in specific pairs. A good example is the AV pair. In the following picture you can see two examples of inter-character spacing, with and without kerning:

With kerning *Without kerning*

You can see that only the kerned image is optically correct because it can compensate for the problem caused by the special form of the V and A characters printed in sequence leaving too much space between the letters.

Metrics Files

Information about a glyph's width is usually located in font files. Kerning information may also be included in the file. In TrueType and TypeTool (.vfb) font formats both metrics and kerning data are located in a single font file. In PC Type 1 (PostScript) fonts the metrics and kerning data are located in separate files.

There are two possible formats for the metrics files that are used with PC Type 1 fonts: AFM and PFM. AFM files (*Adobe Font Metrics*) are text files containing all the metrics and kerning information for a given font. These files are legible as text and can be edited in any text editor. PFM files (*Printer Font Metrics*) are metrics and kerning files used by the Windows operating system. They are binary files and cannot be read without special utilities. AFM files are a standard format for the exchange of metrics information for PostScript fonts. This information can be read directly by several operating systems and programs.

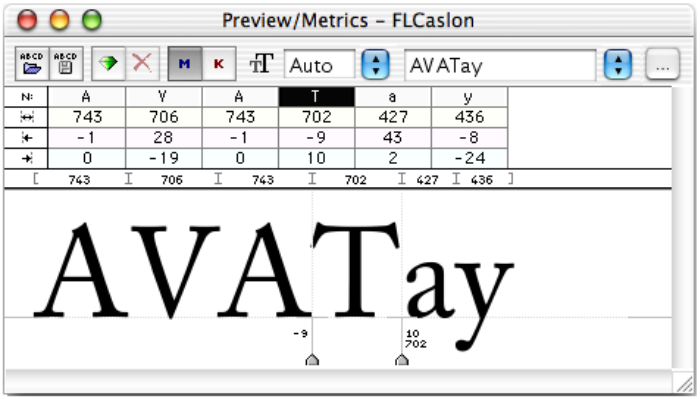
On the Macintosh information about font metrics is stored in the font suitcase data structure and is handled automatically by the Type Manager.

TypeTool can import and export metrics and kerning information in any of these formats.

Metrics Window

TypeTool has a special window where you can edit the metrics and kerning information. It is called the Metrics Window.

To open the Metrics Window select the **New Metrics Window** command in the **Window** menu. The Metrics Window will appear:



The Metrics Window consists of several parts:

- A toolbar with controls for selecting a mode for the Metrics Window, importing and exporting metrics files, automating metrics or kerning generation, a string for metrics or kerning editing and other commands.
- The table that includes “live” digital information about metrics or kerning for the current glyph or glyph pair.
- The editing area where the edited string with controls appears.

There are three possible modes of the Metrics Window: **Preview**, **Metrics** and **Kerning**. The default is the **Metrics mode**.

Use these two buttons to change the mode of the window. Press the **M** button to switch to the metrics mode (glyph metrics are editable but kerning is not), **K** button – to the kerning mode (kerning is visible and editable) and release both buttons to return the window to the Preview mode.

In the **Preview** mode you cannot edit the metrics or kerning data of the font. You can only preview the sample string and look at the glyphs' alignment.

To change editing modes you can also **Ctrl**-click the editing area and select a mode for the Metrics window in the popup menu.

Metrics Window Toolbar

This is a simple list of all buttons available on the toolbar:

	Opens a metrics file (PFM, AFM or suitcase format)
	Saves a metrics file in PFM or AFM format
	Opens the Automatic Metrics or Automatic Kerning generation dialog boxes.
	Opens the Reset Kerning dialog box
	Switches the window to Metrics mode
	Switches the window to Kerning mode
	Opens the Preview Options dialog box

Selecting a String for Previewing or Metrics Editing

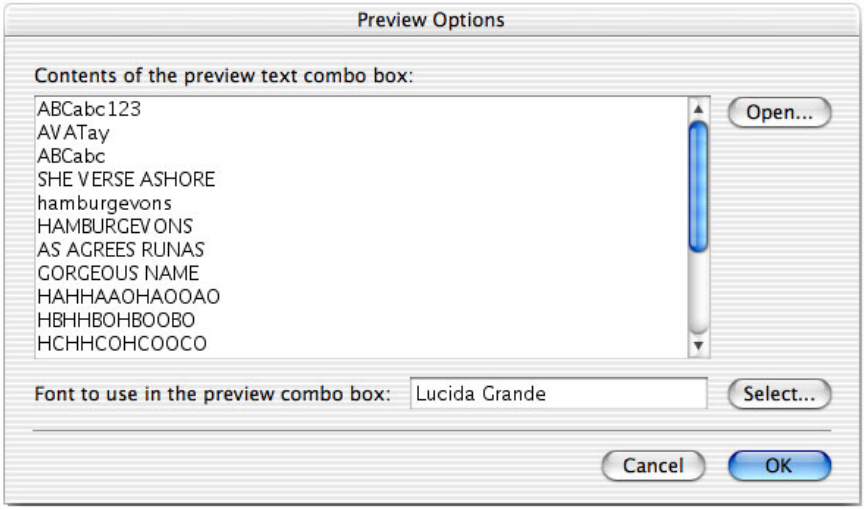
You may want to enter a string of characters to preview or edit their metrics or kerning. The easiest way to do this is to enter the string in the editing field of the Metrics Window and press **Enter**:

Or, you can open a list of predefined strings and select one of them for editing:

By pressing the **Cmd-PgDn** and **Cmd-PgUp** keyboard shortcuts you can browse the strings quickly.

All strings that appear in the list box are read from the Preview.txt file located in TypeTool/Data folder. This file is an ordinary text file and can be edited in any text editor to add your own strings.

You can edit the contents of the strings list in the Preview Options dialog box. **To open the dialog box**, click on the ... button:



You will find an editing field containing all the strings that appear in the list box. Edit the text or use the **Open…** button to fill it from an external text file.

If you are working with a non-Latin font you may want to select the font that is used in the sample string control and list. Use the editing field below the contents field to enter the desired font name or click on the **Select…** button to select the font with the standard font selection dialog box.

Navigating in the Sample String

Use the ◀ and ▶ buttons on the Tools toolbar **to move to the previous or next glyphs in the string**. These buttons are useful if one of the glyphs has a negative width value.

You can also use the **Page Up** and **Page Down** keys on the keyboard to navigate in the sample string.

To scroll the window you can press the space bar and scroll the Metrics window with the Hand tool.

Entering Special Characters

Some glyphs are not represented by the ANSI characters. However, by using special rules you can enter non-ANSI names in the sample string.

To enter a glyph's name, begin with the slash character (/) and finish with a space character or another slash: `/A.small/T.small`

To enter a slash, type it twice (//).

The decimal code of a glyph can follow the slash instead of using its name: `/44`

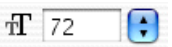
The Unicode index of the glyph can be entered if you use the "\" prefix: `\0445\0448`

To enter a backslash type it twice: "\\".

The easiest way to fill a sample string is **using the drag-drop method**. You can simply drag any glyph from the Font Window and drop it in the Metrics Window and it will be added to the sample string. If you hold the **Cmd** key while dropping the selected glyphs, they will replace the sample string.

Selecting Previewing Size

In the **String Size** combo box:



you can select the size of the previewed string. By default, "Auto" is selected and the size of the previewed text is calculated dynamically, depending on the height of the window.

If the string becomes too large to fit in the window, a vertical scroll bar will appear allowing you to view all the editing areas of the Metrics Window.

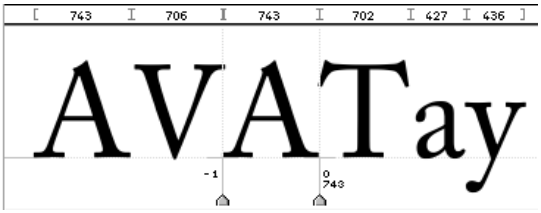
Editing Metrics

By “metrics”, we refer to information about a glyph’s width and sidebearings. In TypeTool you can modify this information either manually or automatically.

To modify a glyph’s metrics you must switch the Metrics Window to the Metrics mode. To do this, push the M button in the toolbar:

or, select the **Edit Metrics** command in the popup menu that appears if you **Ctrl**-click in the editing area of the Metrics Window.

To make the metrics editing controls visible, click on a glyph in the editing field. The sidebearings lines with editing handles will appear at the sides of the glyph:



The numbers at the bottom of the glyph are the left and right sidebearing values and the glyph’s width.

Manual Metrics Editing

To modify a glyph's metrics you can use three methods:

1. Move the sidebearings lines using the mouse.
2. Edit the values in the table area of the Metrics Window.
3. Drag the glyph within the editing area.

To move the sidebearings lines just position the mouse cursor on the line, press the mouse button and drag the mouse. Release the mouse button when you are done.

To drag a glyph within the editing area, position the mouse cursor on the glyphs' image; press the mouse button and drag the mouse to position the glyph inside its width. Press the **Option** key while dragging the mouse to modify the glyph's width.

You can easily **modify the vertical position of the glyph** relative to its baseline. Just press and hold the **Shift** key on the keyboard while dragging the glyph and this will modify its vertical position.

Using the Keyboard

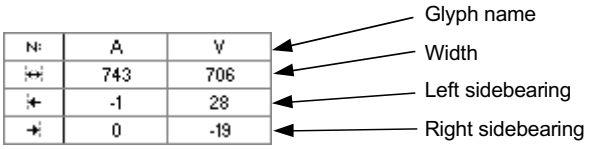
When the glyph is active, you can use the keyboard to adjust the metrics:

Left and right arrow keys	Moves the glyph by one font unit inside the sidebearings without changing the advance width. Hold the SHIFT key to move the glyph by 10 font units.
Cmd+left and right arrow keys	Moves the glyph together with the right sidebearing. This changes the left sidebearing and the advance width. Hold the SHIFT key to move by 10 font units at each key click.
Insert	Changes the current glyph to the next glyph in the font.
Delete	Changes the current glyph to the previous glyph in the font.
Page Up	Moves to the previous glyph in the sample line.
Page Down	Moves to the next glyph in the sample line.
Any character or digit	Selects the character you have clicked as the current character for editing. You can also enter the glyph name if you want to access glyphs that are not assigned to any key combination.

By pressing the **CMD-PGDN** and **CMD-PGUP** keyboard shortcuts you can browse the sample strings quickly.

Using the Metrics Table

In Metrics mode the metrics table consists of four lines:

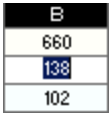


Click on any number in the table to enter an exact value. Just use the keyboard to adjust the number and press the **Enter** key when you are done. The **Esc** key or a left click outside the cell you are editing will cancel the changes.

Use the **up** and **down** arrow keys on the keyboard to move up and down in the table. Use **Tab** key to move right and **Shift+Tab** to move left.

Referencing Metrics

In the table you can use a reference instead of the real numeric value. For example, if you want to set the left sidebearing of the glyph 'B' to be equal to the left sidebearing of the glyph 'D', click on the cell located at the intersection of the B column and the third row:



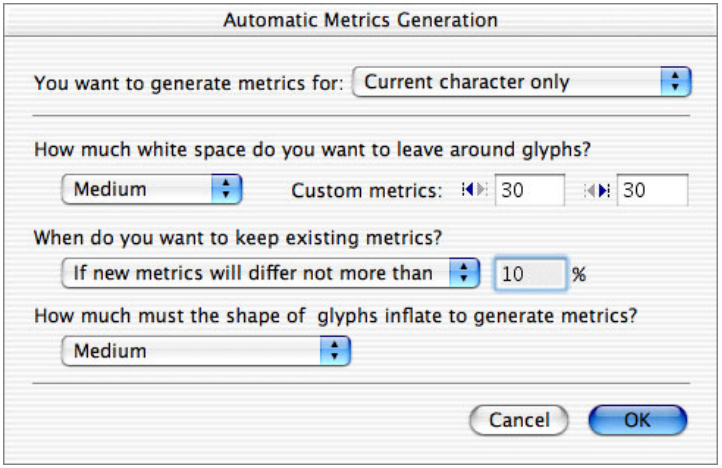
and, instead of the numeric value for the metric, enter "=D". When you click **Enter** key to accept changes, the data will be copied from the source glyph.

Automatic Metrics Generation

TypeTool can automatically define a glyph's metrics using a special algorithm. This algorithm usually produces good results but we recommend manual editing for the best results.

To define the metrics parameters automatically, click on the button on the toolbar or select the **Auto Metrics** command in the context popup menu.

The **Automatic Metrics Generation** dialog box appears:



This dialog box includes two areas: **Area of application** and **Parameters**. In the first area you select the glyph(s) to which the automatic algorithm will be applied.

The possible choices are:

Current character only	This option is available and is the default if any glyph is selected in the editing area.
All glyphs in the current string	This option generates metrics for all glyphs in the current string in the editing area.
Whole font	This forces TypeTool to generate metrics for all glyphs in the font and is not generally recommended. This operation is not undoable.

You can choose the parameters for the algorithm in the **Parameters** area of the Autometrics dialog box. All the parameters are displayed. We recommend that you experiment with various parameters using the autometrics application.

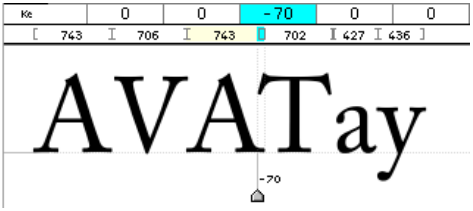
Editing Kerning

To edit kerning data switch the Metrics Window to Kerning mode by pushing the K button in the toolbar:

or, select the **Edit Kerning** command in the popup menu that appears if you **Ctrl**-click in the editing area of the Metrics Window.

To make the Kerning Editing controls visible you must select the pair that you want to edit. Position the mouse cursor on the right glyph of the pair in the editing string and click the mouse button.

You will see the kerning line and handle appear in the editing area:



There is now a blue area in the metrics bar at the top part of the editing area. This means that kerning exists for that pair in the current preview string.

Manual Kerning Editing

To edit kerning manually you must drag the kerning line (or right glyph of the kerning pair) using the mouse. If you **Ctrl**-click the mouse button while dragging the glyph **kerning for that pair will be removed**.

Using the Keyboard

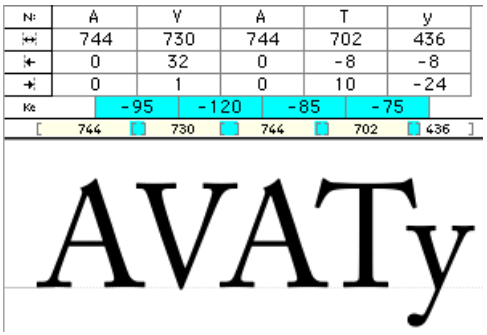
When a glyph is selected in the sample string you can use the left and right arrow keys to change the kerning by one font unit at each key click. Hold the **Shift** key to change the kerning by 10 font units.

Use the **Insert** and **Delete** keys to change the glyph in the string and **Page Up** and **Page Down** keys to move to the previous and next glyph in the string.

By pressing the **Cmd-PgDn** and **Cmd-PgUp** keyboard shortcuts you can browse the sample strings quickly.

Using the Table

When the kerning editing mode is activated you can see all the glyph metrics and pair kerning information in the editing field:



Kerning is displayed on the fifth row in the table and each value is positioned between the glyphs that form the kerning pair. Background color for the kerning value is white when there is no kerning, light blue if kerning is negative (glyphs are shifted toward each other) and yellow if kerning is positive.

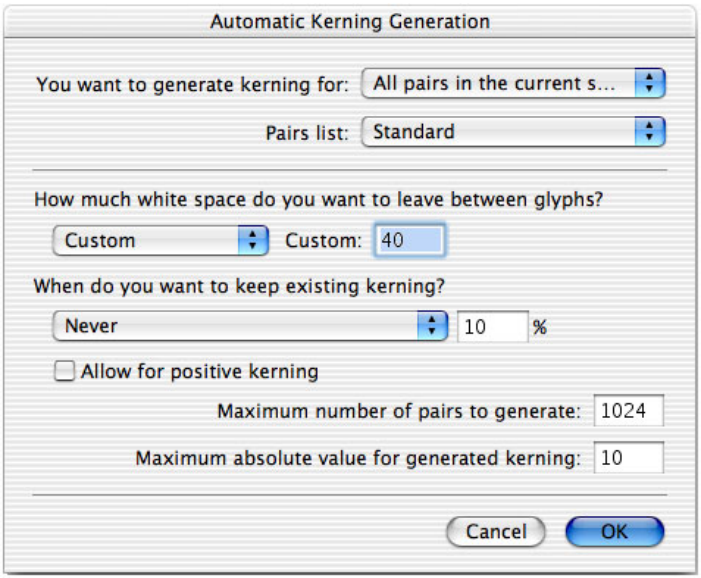
To change the kerning value, click on the kerning row in the table and enter the new value. Click **Enter** key on the keyboard to accept the changes or **Esc** to cancel. Use the **Tab** and **Shift+Tab** keys to select a pair in the sample string.

Automatic Kerning Generation

The easiest way to apply kerning to a font is to use TypeTool's auto-kerning algorithm. This algorithm analyses the shape of the glyphs in the given pairs and automatically kerns them. You can control the pairs list that the autokerning algorithm processes as well as other parameters.

To define kerning automatically press the button on the toolbar, or select the **Auto Kerning** command in the popup menu.

The **Automatic Kerning Generation** dialog box appears:



This dialog box consists of two areas: the **Area of application** and **Parameters**.

In the first area you select the pairs for which the algorithm will compute kerning values. You can choose between **Current pair only** (available if one of the pairs is selected in the editing area), **All Pairs in the current string,** or **All Pairs in the following list**.

The second popup menu allows you to generate kerning for all the pairs located in a special list file. The list files are stored in the TypeTool/Kerning folder. You can create your own kerning pair files or use files placed there by TypeTool's installer.

The **Parameters** area lets you customize the autokerning algorithm. The most used option is: **How much white space do you want to leave between glyphs?** This controls how close the glyphs will be moved together while computing kerning in the pair.

The **Allow for positive kerning** check box lets the autokerning algorithm produce positive kerning in pairs. Positive kerning moves glyphs apart from each other. Positive kerning is usually not recommended but there may be occasional circumstances where it is needed.

If you want to save the existing kerning the popup menu lets you control the disposition of the existing (imported or manually created) kerning pairs. You can replace existing pairs by automatically generating new ones, keep them unchanged, or select the condition mode.

The **Maximum number of generated pairs** and **Maximum absolute value for generated kerning** options control the possible number of automatically created pairs and the maximum normal (negative or positive) kerning value.

Resetting Kerning

To remove the kerning information for some glyphs or for the entire font you must use the *Reset Kerning* feature. To open the Reset Kerning dialog box press on the or select the **Reset Kerning** command in the popup menu.

The **Reset Kerning** dialog box appears:

Reset Kerning
What exactly do you want to do?
Reset kerning for the current pair
Reset kerning for all pairs in the string
Reset kerning for all glyphs in the string
Completely reset kerning in the current font
Cancel
OK

This dialog box includes options that control kerning removal.

Available options are:

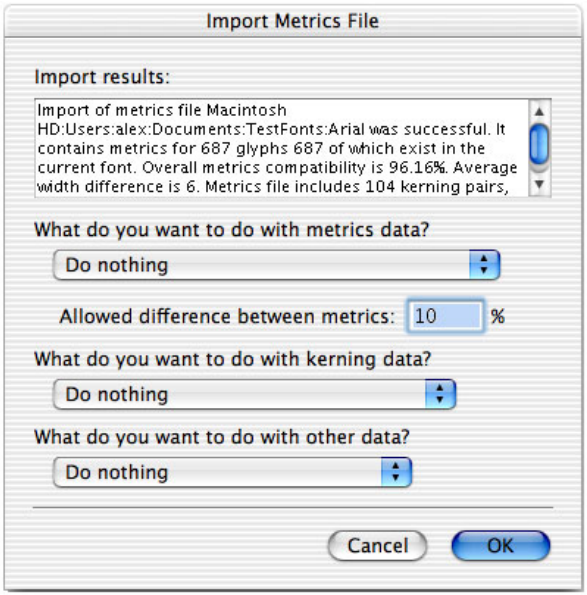
Reset kerning for the current pair	This is the default if a pair is selected. Removes kerning for that pair only. You can get the same result by Ctrl-clicking while editing kerning in the current pair.
Reset kerning for all pairs in the string	Default if no pairs are selected. Removes kerning in all pairs that exist in the current string.
Reset kerning for all glyphs in the string	Removes kerning in all pairs that include glyphs in the current string.
Completely reset kerning in the current font	Removes all the kerning pairs available in the font. Because this is not an undoable operation, the warning dialog asks you to save the current metrics and kerning data in the temporary file.

Opening Metrics Files

TypeTool allows you to import metrics and/or kerning information into the current font. Using this feature, you can create metric and kerning information once and use it in several similar fonts.

To import a metrics file into TypeTool click the button on the Metrics window toolbar. You will see the standard Mac OS Open File dialog box. Select the metrics file that you want to import (in PFM, AFM or suitcase format) and press the **Open** button.

The **Import Metrics File** dialog box appears:


Import Metrics File
Import results:
Import of metrics file Macintosh HD:Users:alex:Documents:TestFonts:Arial was successful. It contains metrics for 687 glyphs 687 of which exist in the current font. Overall metrics compatibility is 96.16%. Average width difference is 6. Metrics file includes 104 kerning pairs,
What do you want to do with metrics data?
Do nothing
Allowed difference between metrics: 10 %
What do you want to do with kerning data?
Do nothing
What do you want to do with other data?
Do nothing
Cancel
OK

The topmost control contains a legend describing the metrics file that you are importing and its compatibility with the current font.

The options in the **Parameters** area let you select various metrics importing options:

What do you want to do with the metrics data:

Do nothing	Do not import metrics data from this file.
Replace all metrics in the current font	Import all metrics data (glyphs' widths and sidebearings) and replace the metrics data in the current font. We recommend that you use this option only if your font is very similar to the metrics file that you are importing.
Replace all metrics that are close to current	Replace only those metrics records that are similar to the imported metrics. The **Allowed difference between metrics** option controls the possible difference.
Replace metrics that are thinner than in the current font	These options are obvious.
Replace metrics that are wider than in the current font	

What do you want to do with the kerning data:

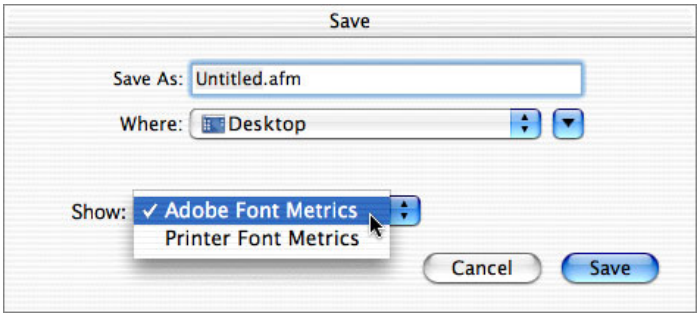
Do nothing	Do not import kerning data from the metrics file.
Completely replace kerning data in the current font	Remove all existing kerning pairs and replace them with pairs imported from the metrics file.
Add imported kerning data to the current font	Leave the existing kerning pairs unchanged but add new kerning pairs from the metrics file.
Add new kerning pairs but autokern them	Import information about the glyphs that form each kerning pair in the metrics file and apply an autokerning algorithm to these pairs.

The **What do you want to do with other data?** option controls the font header importing option. TypeTool can import the Font Info data from the metrics file and replace the current font info data if the **Replace this data in the current font** option is selected.

Saving Metrics Files

When you export a font file in PC Type 1 font format the metrics files (in AFM and PFM formats) are automatically written. The TrueType font format includes all metrics information so it is not necessary to export additional files. Same with Macintosh fonts.

However, if you want **to export a metrics file alone**, you can always do so by using the Metrics Window. Just press the button on the Metrics Window toolbar. The Standard Save File dialog box appears:



Select the destination format (Adobe Font Metrics or Printer Font Metrics for single master fonts), and the destination folder. Enter the file name and press the **Save** button to save the metrics file.

Printing a Sample String

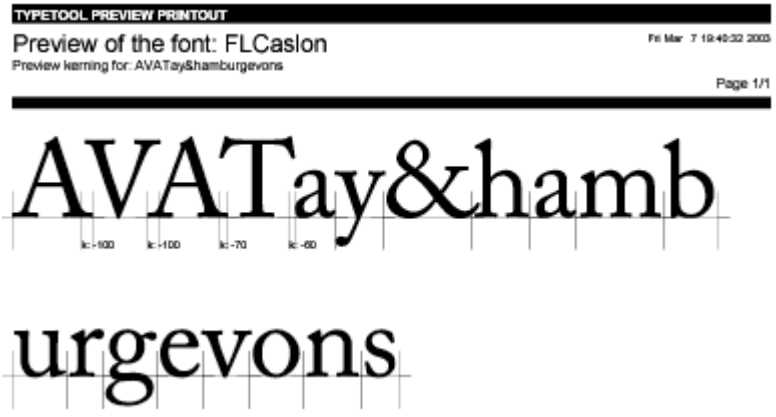
While you are in the Metrics Window you can print sample strings with or without metrics and kerning information. To do so select the **Print** command in the **File** menu.

You will see the standard Print dialog box where you select the printer and modify printing options. Press **Print** to begin printing.

TypeTool will print the string that is currently selected for editing or previewing in the Metrics Window. The size of the printed string will be the same that is selected for previewing on the screen. The string may occupy more than one line or even more than one page on the paper if necessary.

If metrics or kerning editing mode is selected additional information will be printed below the baseline: in metrics mode – the widths of the glyphs and the glyphs' sidebearings; and in the kerning mode – the pair kerning values.

In the metrics editing mode kerning will not be applied to the printed string.



Transformations

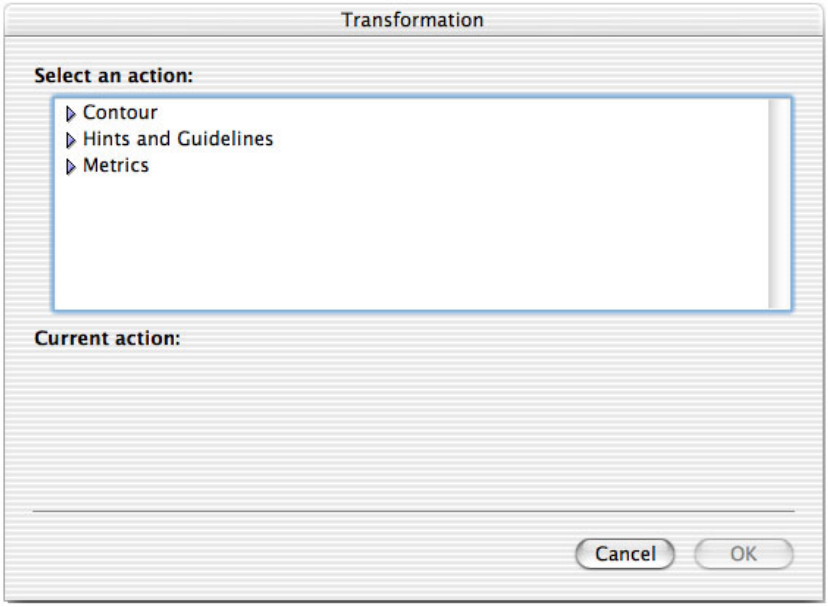
In TypeTool you can transform glyphs in many ways. You can edit glyphs and glyphs' metrics manually using the Glyph and Metrics windows described in previous chapters. Or you can use TypeTool's transformation actions to edit glyphs or metrics automatically. Transformations may be applied to one glyph or to a range of glyphs selected in the Font window. All the actions are carefully designed and always produce high-quality results that usually do not require manual control or correction.

In this chapter we will show you how to use the transformation actions and give a detailed description of each available action.

6

The Transform Dialog Box

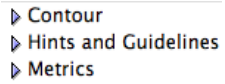
To apply transformations you have to use the Transform dialog box. It is accessible from the **Tools** menu while the Font or Glyph window is active. Select the **Transform** command from the **Tools** menu and you will see a dialog box:



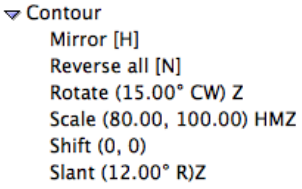
By default the box is empty and the dummy transformation action **Do not transform a glyph** is selected in the actions list.

If you open this dialog while the Glyph window is active the transformation filter will be applied only to the glyph currently open. If you open it while the Font window is active then the transformation will be applied to all selected glyphs. Transformation of many glyphs is not undoable.

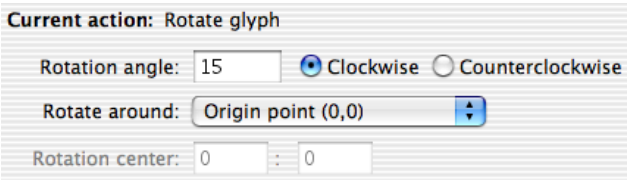
To select an operation use the *Select an action* list:



Expand one of the categories to see all the actions:



Select an action and you will see an action customization panel appear below the list:



The contents of the customization panel depend on the action selected.

After you select a transformation action and set its options, press the **OK** button to begin transformation. If you are transforming more than one glyph a warning message will appear telling you how many glyphs you will modify and asking you to confirm the transformation. Transformation of many glyphs is not undoable, so it's a good idea to save your font before taking this action.

Below you will find a detailed description of each available transformation action.

Transformation Actions

There are three groups of actions:

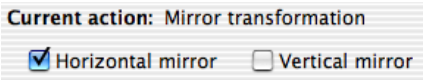
Contour	The outline of a glyph is transformed.
Hints and Guidelines	Actions that are concerned with hints and links.
Metrics	Metrics information is transformed.

Outline Transformation

Here is a list of all the outline transformation actions:

Mirror	Mirrors the glyph vertically or horizontally
Reverse all	Reverses the outline direction of the glyph
Rotate	Rotates the glyph
Scale	Scales the glyph proportionally or non-proportionally
Shift	Shifts the glyph's outline
Slant	Slants the glyph

Mirror



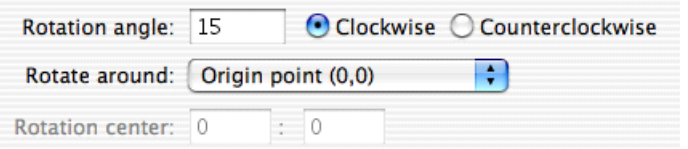
Here is the result of this simple transformation:

The letters in the word "Sample" were mirrored horizontally and the letters in the word "text" – vertically.

Reverse all

This action reverses the direction of all the contours of the glyph's outline. It can also automatically detect the direction of contours and correct them if necessary. Just select the option that you want.

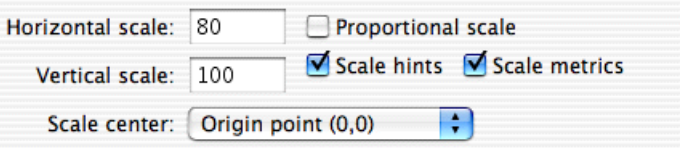
Rotate



This transformation action simply rotates glyphs. You can set the rotation angle, the position of the center of rotation and the direction of rotation.

You can rotate glyphs around the zero point, around the center of the glyph's bounding box or you can specify a point that will be used as the center of rotation. Here is an example of the same rotation transformation around different center points:

Scale



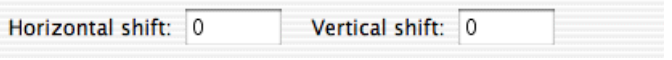
This action lets you scale your glyphs proportionally or non-proportionally. Enter your desired vertical and horizontal scale factors in the edit fields. Switch on the **Proportional scale** option to keep the vertical and horizontal scale factors the same.

Switch off the **Scale hints** option to avoid scaling hints along with the glyph's outline. Scaling hints' width is not always precise so if you scale hints with the outline you sometimes find that some hints now miss the stems that they were supposed to hint. We recommend manual editing of hints after scaling applied.

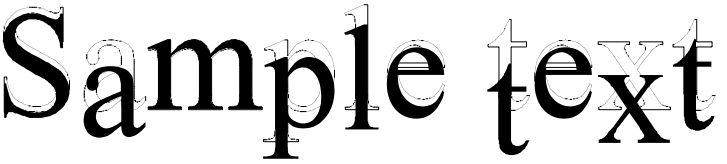
Here is an example of this transformation (the letters of the word "Sample" were scaled 80% horizontally and the letters of word "text" were proportionally scaled to 120% of original size):

Sample text

Shift

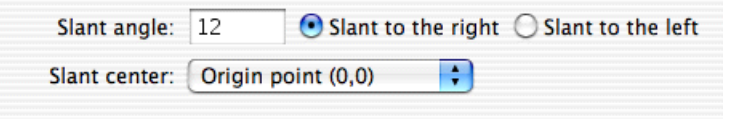


This action shifts the outline of the glyph in the vertical and/or horizontal direction. Here is a sample of a font with some glyphs shifted in the vertical direction:



You can also shift glyphs in the vertical direction in the metrics mode of the Metrics window: hold the **Shift** key and drag the glyph.

Slant



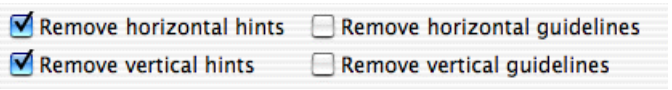
This action slants glyphs. It's the quickest way to make an oblique version of your font. Just apply this transformation to all the font's glyphs and correct the Font Info settings to let the operating system know that this font is now oblique.

Here is a sample of Slant transformation ("Sample" is slanted 12 degrees to the right and "text" is slanted 30 degrees left):

Hints and Guidelines Transformation

Hints transformation actions let you automate hinting actions.

Remove hints/guides

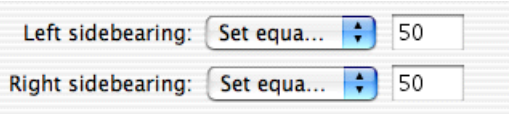


Use this action to remove hints or guidelines in selected glyphs. This action is the equivalent of the **Remove Hints** and **Remove Guides** commands that were described in the “Glyph Window” chapter.

Metrics Transformation

This transformation lets you automatically increase or decrease a glyph's sidebearings and width.

Set Sidebearings



Use this action to change the sidebearings' values of the glyphs. You can set new values for the left or right sidebearings or change these values by entering the amount in font units. So if you think that your font needs some more white space, just select this action, choose the **Increase by** option in the popup menus and enter the value by which you want to increase the sidebearings.

Here is a sample of increased glyph widths:

Index

A

Actions, 83, 186
  hints and guidelines, 198, 244
  list, 243
  metrics, 244
  outline, 186, 244
Adobe Illustrator, 174, 213
Adobe Standard Encoding, 55
AFM, 9, 95, 220, 237, 239
AI, 213
ANSI, 55, 225
Arabic, 125
Ascender, 118, 121
ATM, 90, 97
ATSUI, 90, 97
Autohint, 44, 98
autokerning, 9
Automatic Kerning Generation, 234
Automatic Metrics Generation, 230
Autosave, 70
autospacing, 9

B

baseline, 188, 190, 227
  properties panel, 190
Bezier curves, 44, 141, 161
Bitmaps, 97
Bitstream, 125
blue marks, 141
Break, 164
BTBD, 122

C

Caps height, 118
cell, 48
Character, 14, 52
Character mapping standard, 51
Character Set, 125
  OEM, 125
  ShiftJIS, 125
  Symbol, 125
Characters
  deleting, 74
characters' cells, 47
Codepages, 58, 124
  Double-byte, 61
Codepages mode, 53
Component, 210
  properties panel, 212
Composite, 43, 208
composite characters, 208
connections, 143, 186
Context Menu, 17, 65
continuous, 64
Contours, 139
  drawing, 200
  joining, 164
  merging, 186, 187
  selecting, 170
  smoothing, 145
convert
  curve to vector, 163
copy, 71, 172
Copy characters, 71
Copying characters, 72
Copyright, 111
cubic b-splines, 141
cut, 172
Cyrillic, 55, 58

D

decompose, 43, 44
decomposing, 208, 209
Deleting Characters, 74
Deleting Nodes, 155
Descender, 121
Description
  FontInfo, 111
Designer, 112
Directories, 17
Double-Byte Codepages, 82
Drag-Drop, 38, 103
Duplicate, 172

E

Echo, 150
Edit, 134
Edit Mode, 134
Edit Window
  change a view in, 131
Editing Field, 128
Editing hints, 197
Editing Kerning, 232
Encoding, 15
  table, 54
encoding files
  custom, 56
EPS, 9, 213
Eraser, 134, 156
Export Options, 94
Exporting
  fonts, 84
  glyphs, 213
External Programs, 23
extremes, 186

F

font, 15
Font Family, 15
Font Formats, 84
font height, 138
Font Info, 49
Font Map Panel, 80
Font Metrics
  What are, 218
Font window, 26, 46, 62, 65, 102, 128, 225
  Customizing, 102
  popup menu, 65
  selecting, 64
FontInfo
  Ascender, 118
  Caps height, 118
  Copyright, 111
  Created by, 111
  Creation year, 111
  Descender, 118
  Description, 111
  dialog box, 106
  Family Name, 109
  Font Name, 110
  Full Name, 110
  Italic, 109
  Italic angle, 118
  Menu Name, 110
  Notice, 111
  Revision, 114
  Slant angle, 118
  Style Name, 110
  TrueType Unique ID, 115
  TrueType Version, 114
  Unique ID, 115
  Vendor Code, 115
  Version, 114
  Weight, 109
  Width, 109
  x height, 118
  XUID, 115
FontLab, 84
  2.0, 39
  2.5, 39
  3.0, 39
Fonts, 34
  Multiple Master, 40
  opening, 39
FreeHand, 174

G

glyph, 14
Glyph name, 16
Glyph Toolbox, 134
Glyph Window, 28, 127
  open, 128
glyphs, 51
  exporting, 213
  importing, 213
graphical primitives, 9
green marks, 141
Grid, 136, 191
guideline, 195
  properties panel, 195
Guidelines, 136, 192
  editing, 193

H

hint
  properties panel, 198
hinting
  character-level, 196
  font-level, 196
Hints, 136, 196
  Editing, 197

I

Illustrator, 174, 213
Importing
  glyphs, 213
Index, 47
INF, 95
Intersection, 186
ISO Latin 1, 55
Italic angle, 118

J

Join, 164

K

Kerning, 218, 219
  automatic generation, 234
  editing, 232
  manual editing, 232
  mode, 33
  resetting, 236
Keyboard, 21
Knife, 134, 164
Knife tool, 155, 157

L

License, 113
Line Gap, 121
Local Toolbar, 128

M

Mac, 120
Macintosh, 11, 13, 18, 20, 39, 55, 57, 84, 86, 92, 121, 125, 220
Magic Wand, 170, 171
Mapping Encoding, 54
master font, 40
measure, 160
Menu, 16
Merge, 186
Meter, 134, 159
  bar, 159
Metrics, 188
  assistants, 9
  automatic generation, 230
  editing, 226
  files, 220
  manual editing, 227
  mode, 32
  Opening Files, 237
  properties panel, 190
  Saving Files, 239
Metrics window, 31, 128, 221
  Kerning mode, 232
  Metrics mode, 226
  modes, 221
  Preview mode, 33, 221
Mirror, 245
MMM, 239
Mode
  VectorPaint, 200
move, 177
Move characters, 67

Moving Nodes, 148
Multiple Master, 39, 40, 84

N

Names mode, 53, 55
node, 141
  properties panel, 168
Nodes
  Deleting, 155
  Inserting, 157
  Moving, 148
  selection, 170
Non-nodes editing, 152
Notice, 111

O

Operation
  Component Positioning, 210
  Transform, 175
Outline Actions, 186

P

Panel
  Font Map, 80
  Transformation, 181
Panels, 34
PFA, 37, 39
PFB, 37, 39, 85
PFM, 9, 95, 220, 237, 239
Popup Menu
  Guideline and Hint, 198
PostScript, 9, 55, 85, 115, 213, 220
preview
  mode, 33
properties panel
  baseline, 190
  component, 212
  guideline, 195
  hint, 198
  metrics, 190
  node, 168
  selection, 180

R

red marks, 141
rename, 78
Reverse all, 245
rotate, 177, 246
Rulers, 30, 129

S

Sample String
  navigating, 224
Scale, 247
Searching, 76
selection
  scale, 177
selection
  move, 177
  properties panel, 180
  rotate, 177
  skew, 177
  slant, 178
Set Sidebearings, 250
Shift, 248
Show
  Connection mode, 146
  Control vectors, 146
Show Nodes, 146
slant, 178, 193, 248
Slant angle, 118
Smooth, 103
Snap, 135, 186
splines, 44
startpoint, 141
suitcase, 37, 39, 84, 86, 87, 125
supercurve, 142
Symbol, 55

T

Tool
  Add corner, 134
  Add curve, 134
  Add tangent, 134
  Bezier curves, 134, 161
  Edit, 134
  Edit, 134, 155

Eraser, 134, 156
Knife, 134, 155, 157, 164
Magic Wand, 170, 171
Meter, 134, 159
Zoom, 131
Toolbar, 18, 19
Edit, 130
Export Suitcase, 88
Font Map, 80
Glyph Window, 128
Metrics Window, 32, 222
Show Layers, 136
Standard, 26, 172
Transform, 175
Transformation Panel, 181
Transformations, 241
TrueType, 9, 37, 39, 84, 85, 114, 115, 220, 239
TrueType curves, 141
TrueType hinting, 9
TrueType instructions, 44, 98
TTF, 37, 39
Type 1, 9, 37, 84, 85, 114, 115, 196, 220, 239
Type 1 curves, 141
Type Manager, 115
TypeTool
user interface, 10, 20
TypeTool, 9
Typo Ascender, 121
Typo Descender, 121
Typo Line Gap, 121

U

Unicode, 43, 52, 57, 99, 225
Codepages, 58
Consortium, 57
duplicating, 73
Unique ID, 115
UPM, 117, 137

V

VectorPaint, 9, 200, 205
Brush, 201, 203
Contour, 201, 202
Ellipse, 201, 206
Line, 201, 206
options, 203
Pen, 201
Rectangle, 201, 206
Text, 201, 207
Vendor Code, 115
Vertical Metrics, 199
VFA, 39, 85
VFB, 37, 39
Visual Ascender, 133
Visual Descender, 133

W

WinAscent, 121
WinDescent, 121
www.fontlab.com, 36, 84

X

x height, 118
XUID, 115

Z

zoom mode, 131